RELATION BETWEEN PROCESS AND SERVICE LOGICS

Milan Mišovič, Jan Turčínek

Abstract

MIŠOVIČ MILAN, TURČÍNEK JAN: *Relation between process and service logics.* Acta Universitatis Agriculturae et Silviculturae Mendelianae Brunensis, 2013, LXI, No. 2, pp. 411–416

It is generally accepted that the process control of a small and medium-sized manufacturing business enterprise is the foundation of high quality care of firm's business processes. Any business process is seen as an indivisible sequence of activity steps designed to perform complex business activities. In its statutory documents the company should have concise descriptions of at least the main processes, along with their contexts in a given department of the company and the employee position.

The main business processes, of course many others, are not immutable, on the contrary, they are very often changing. Many processes occur, others are modified others disappear as antiquated and useless to support strategic business objectives. All this is a consequence of the firms' effort needed to maintain competitiveness in the harsh and dynamic consumer market.

Business processes are not isolated, many of them are part of a relatively large process chains, so-called enterprise services, see (Erl, 2005). The discipline of Software Engineering responded to the possibility of consolidating enterprise functionality with enterprise services with the method SOA (Service Oriented Architecture) leading to new applications for enterprise information systems.

In contrast to business processes, business services are still not sufficiently recognized in the statutory documents of enterprises. Informaticians, producing software applications for enterprise information systems, must draw on company management knowledge relating to the general context and processes together with management to prepare business services. There are therefore more relevant questions based on the emergence of corporate services and information modeling in the discipline of Information Engineering. Acceptable responses are not included in a lot of publications or in publications of the doyen of SOA Thomas Erl, see (Erl, 2006) and thus the proposed SOA paradigm suffers from the same problem.

The present article tries to give an answer to those questions and show the relevant theoretical basis for finding service solutions of business process logic. Furthermore, this article wants to show possible conversions of known methods of process analysis of Information Engineering disciplines, such as the method Eriksson – Penker Business Extensions, or the method ARIS by prof. Scheer, into the platform of enterprise services.

process logic, process threads, enterprise services, service logic, SOA (Service oriented Architecture), process diagram, theory logic

METHODS AND RESOURCES

1 Concept of any theory logic

From the formal mathematical logic is known that each theory is composed of four components:

- set P of all objects, called the carrier, over which theory is defined,
- $\Im$ language of theory, in which we write operations over elements of media theory,
- $\mathcal{L}$ logic of theory,
- A finite set of axioms of the theory.

Often, for theory $\mathcal{T}$ is very often used a formal notation $\mathcal{T} = (P, \mathfrak{I}, \mathcal{L}, A)$.

Mathematical logic understands the logic of theory as an analytical apparatus over certain objects carrier theory, which the carrier elements, respectively over subsets carrier assigns a new quality or transforms elements of the carrier on new analytical objects.

In practice, we meet with many interpretations of logic theory. Select one typical case involving propositional calculus. Based on this, of course respecting differences, we will try to bring the concept of business process logic, and then also the logic of enterprise services.

Example 1

Propositional calculus is a theory with the empty set of axioms, ie. $A = \phi$. The logic of propositional calculus is given by the following definition 1.

Definition 1

$\mathcal{L}$ logic of propositional calculus $\mathcal{T}$ is called the set $\mathcal{F}$ formulas along with some relationships $\vdash$ between $\mathbb{P}(\mathcal{F})$ and $\mathcal{F}$ that is called as a correct logical consequence. Logic $\mathcal{L}$ structure can be considered as $\mathcal{L} = [\mathbb{P}(\mathcal{F}) \times \mathcal{F}, \vdash, \mathfrak{R}]$, where $\vdash$ is certain binary relation on $\mathbb{P}(\mathcal{F}) \times \mathcal{F}$ correct logical consequence. The set $\mathfrak{R}$ is the final set of production rules of a correct logical consequence. $\Gamma \vdash \mathbf{a}$ notation then indicates that to the set of formulas Γ, formula $\mathbf{a}$ is the correct logical consequence.

2 Enterprise business logics

In the last two decades we have seen major changes in the view of the business enterprise. It introduced the new concept of enterprise architecture, began to talk about the business process logic and business logic services as two business logics of commercial production company. It is interesting that the both logic**s** do not go against each other. It is recognized that the logic of business services is a continuing of business process logic. In this sense there is introduced also the idea of the service for the enterprise business logic in the article (Mis – Rab, 2012). But in this article there is a significant formally different approach to the roles of both logics based on the knowledge of formal logic. On the other hand, in this contribution a different role in relation to business functionality is significantly assigned to both logics, although the aforementioned preserve continuity. To the process logic there is assigned the task to capture the default context among processes and the resulting data, organizational units, products, documents, information, knowledge, events, employee position, destination, ... and on the other hand, there is not assigned a possibility to produce process threads.

Unlike process logic, to service logic there is just assigned the ability to form – on the basis of certain rules – certain process threads that consider business services and also to assign the ability to assume the basic properties of process logic.

It turned out that the enterprise-based business services functionality of computerization gives more options of distribution, dynamism for programing of enterprise information system components – enterprise services, see (Woods – Mattern, 2006). This is also fully reflected particularly in the SOA architecture.

3 Concept of Enterprise process logic

P be the set of all business processes, D the set of all organizational business units, D_i input data set of all processes, D_0 the output data set of all processes, E the set of all events and $\mathfrak{R}$ set of procedural rules. Then we can write the business process logic by means of this notation:

$$\mathcal{L} = [P \times P, P \times D \times D_i \times D_0 \times E \times \mathfrak{R}].$$

In short, the process logic associates the given process with the other processes but does not produce any new compact units of them. Process logic refers to a corporate wide area, in which the given process is alive. Of course, the context of the process can be expanded on its surroundings and thus previous notation is merely a pattern of how to proceed.

Process logic is fully reflected in many process modeling methods. Just note the methods of Eriksson – Penker Business Extensions, or the method ARIS by prof. Scheer. Displayed contexts of business processes in these two methods are very broad, and stratified into several levels. The next example illustrates a company and its process diagram on the first level.

Example 2

Our company has the main objective of interest a mediation group of discounts for customer services and consultancy in the field of optimization of costs for various customer services for households. Mediated customer services include mobile services operators, car insurance and energy.

In the pet business there have been identified five fundamental processes that form the backbone of the enterprise functionality on the main business objective, which is profit. They are:
- Questionnaire Survey
- Targeted range of customer services
- Technical support of customer service
- Administration orders,
- Activation of customer service by supplier.

The above processes are the main components of the structure of business processes with regard to the strategy pursued by the company management. These processes represent the business processes highest level. Accounting process is not taken into considerations because is carried as an external service.
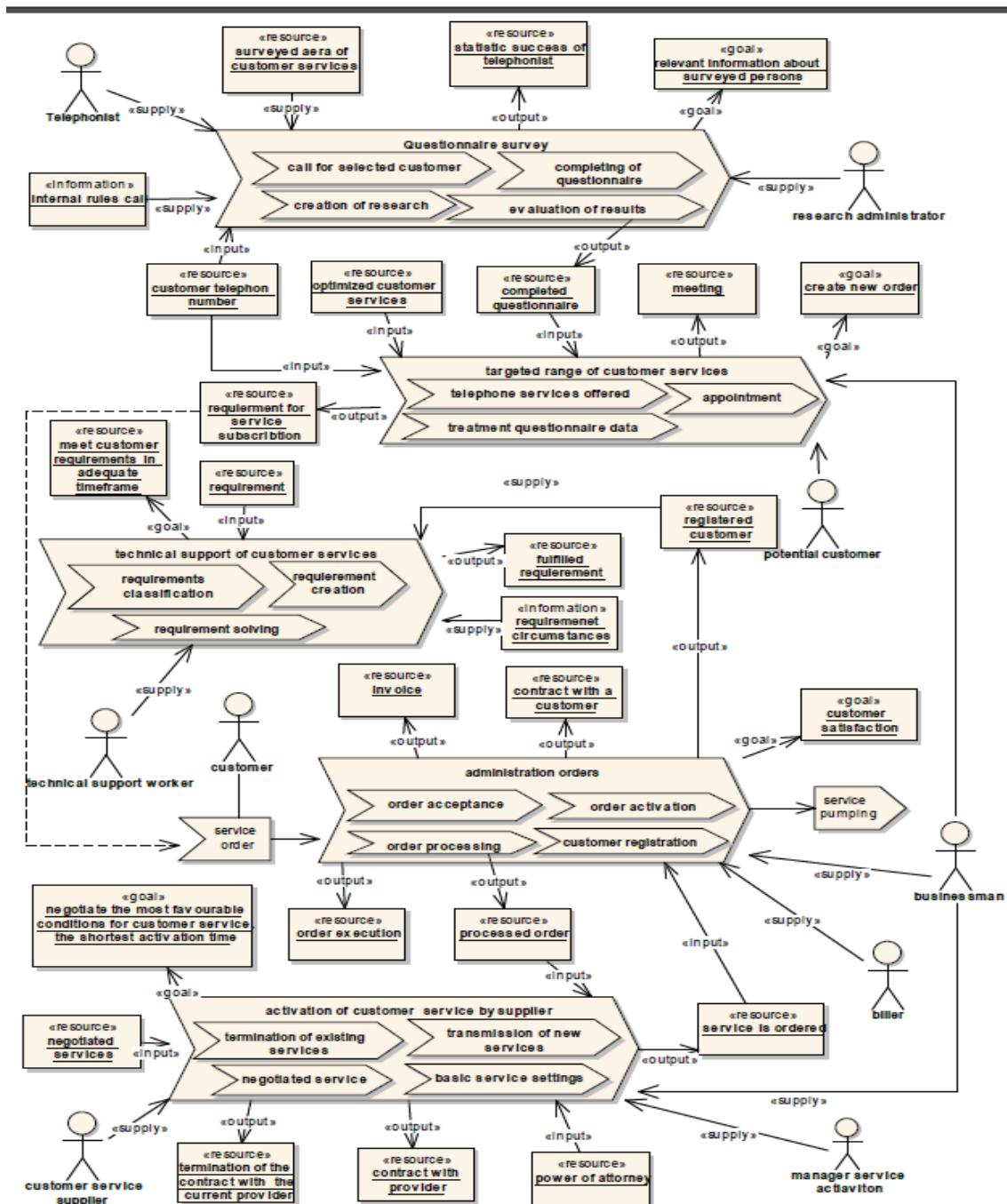
1: *Scheme of the main enterprise processes, according to Eriksson – Penker Business*
Extensions method. Taken from (Tom, 2012).

4 Concept of Enterprise service logic

Let us look at the concept of business service logic in terms of Knowledge engineering and formal logic. It is not fantasy to think that just a set of mutual associations and business processes is the foundation of the business service logic $\mathcal{L}$' This can be understood as a set of production rules, based on the identified associations between processes, producing coherent process threads of them. We will regard these produced threads as enterprise services.

But the basis of any business service logic, however, is the association between two business processes with the possibility of generalization to a finite number of processes.

Let us explain now the concept of service logic.

Definition 2

Be S the set of enterprise services, A set of all associations between business processes, $\mathfrak{R}$ set **of** all the rules for enterprise services. Given what

has been said, we can write the logic of enterprise services following notation

$$\mathcal{L}' = [\mathcal{L}, \mathbb{P}(P) \times S \times \mathbb{P}(A) \times \mathbb{P}(\mathfrak{R})].$$

Cartesian product $\mathbb{P}(P) \times S \times \mathbb{P}(A)$ documents the creation of process threads – a set of business services with a subset of the processes $\mathbb{P}(P)$ based on the association sets $\mathbb{P}(A)$. The newly created set of processes – business service is tied to a set $\mathbb{P}(\mathfrak{R})$ limit the procedural rules of the participating processes. They remain as the limiting factor for the functionality of the newly created business services. By $\mathcal{L}$ symbol the business service logic follows the process logic of business, since the context of sub-processes becomes the context of the new business service.

If we simplify the concept of Definition 2 in shape $\mathcal{L}' = [\mathcal{L}, P \times S \times A \times \mathfrak{R}]$, then we understand each process as a service, but in practice is this case not fully effective.

Another example is the concept of so-called native service logic that is compiled on the basis of knowledge of associations between business processes.

Native transformation basis is the closest to the company management process thinking. It is characterized by the fact that it utilizes the results of process logic that produces process threads, much more complex entities than individual processes. This transformation is very often used in practice.

The specific semantic systems based on different types of associations between business processes beyond the concept of native business process logic, give the possibility of very specific enterprise services. For example, services based on application of artificial intelligence methods for business management, enterprise production, etc.

4.1 Algebra of Enterprise service logic

In the case of native understanding of the concept of enterprise services we can support business process management thinking by several operations in so-called service logic algebra.

Definition 3

Be p, q two different business processes. Be A a set of all associations between business processes identified by the company management. Let A' be one of the subsets of A. If for processes p, q the association A' is valid, which we write $p \, A' \, q$, then the ordered pair (p, q) is a new thread produced by the business service logic.

Although the (p, q) process thread is a significant result of the business process logic production, the company management knows and can manage process threads far higher than 2. To produce such process threads we suggest – for the business process logic – three operations **Create prefix**, **Create suffix** and **Create insert** that can freely expand members of process threads, thus increasing the number of process threads dimension.

Definition 4

Be (p, q) a process thread. Let the associations A_1 be valid for processes r, p with production rules leading to the thread (r, p), then the operation **Create prefix** is able to create a thread (r, p, q).

Be (p, q) a process thread. Let the associations A_2 be valid for processes q, r with production rules leading to the thread (q, r), then the operation **Create suffix** is able to create a thread (p, q, r).

Be (p, q) a process thread. Let the associations A_3 **be valid for processes** p, r with production rules leading to the thread (p, r) and A_4 associations between the processes r, q with production rules leading to the thread (r, q), then the operation **Create insert** is able to create a thread (p, r, q).

These operations **Create prefix**, **Create suffix** and **Create insert** are possible interpretations of specific production rules of the associations A_1, A_2, A_3 and A_4.

Understanding the associations between business processes is far from completed business management knowledge. There is known that a research of interrelations between process threads and their orchestration inside the system based on them is not completely finished. On the other hand, the rugged and dynamic consumer market makes any company introduce new business processes, and there is always necessary to find their position in the already established process threads, or to create a new process thread for them. There is obvious that the system of threads is very much alive, and is flexibly modified. Given that our goal is a more advanced platform of enterprise activity, a business service platform, we will not deal with process threads problems consistently.

Although the information modeling of individual processes can use the UML language activity diagrams, process threads should be modeled by a more transparent and widely more capable technology. The appropriate technology is a proven workflow, i.e. network graphs on the platform of features and possibilities of Petri nets.

5 Conversion of Eriksson – Penker process diagram to enterprise service schemes

In publications on SOA there are very rarely given independent analytical methods for finding business services. More frequent are methodological procedures, explaining what types of business services we can meet in enterprise (business, decision–making, ...) and how to interact with the management company at their destination.

In the previous text, we have spoken about the strong connection between the process logic and business service logic, i.e., we know that the service logic of enterprise is directly derived from process logic. Enterprise services divide business functionality into modules, thus individual flexible units. In other words, business functionality expressed on business services in any way does not suppress the essential role of business processes, but rather points to the possibility of introducing

a quality architectural model of enterprise functionality. This will be reflected in the target software of enterprise information system. There is generally known next statement of Thomas Erl *a good piece of knowledge of business processes is the basis for recognizing good enterprise services*.

Many authors of publications about SOA also indicate the possible use of high-quality analytical methods for modeling business process logic. It is not the aim of this contribution to debate on these methods. But we will try only to show how easy is a conversion of process model obtained by Eriksson – Penker Business Extensions models to business services schemes and their links with interfaces.

The process diagram obtained by Eriksson – Penker Business Extensions has a tier character. The diagram can have several levels. Process modeling begins by the highest levels of enterprise processes. In Example 1, it is the level that is composed of the processes Administration orders, Activation of customer services, Technical support, Questionnaire Survey, Targeted range of customer services.

Now we describe relevant conversion steps.

We convert this highest level diagram to enterprise services, and we use process names for the names of enterprise services. These services do not contain any reciprocal links yet. Services are considered as separate modules.

For each of the resulting enterprise services we will create the service interface - interface of entities in versions offered and required. We can find a simple algorithm that can see the context through selected business services and can easily assemble the interface. For example, the service *Questionnaire survey* has its interface as follows:

Required:
● Surveyed area of customer services.
● Customer telephone number.
● Internal rules call.
Offered:
● Statistics success of telephonists.
● Relevant information about the surveyed persons.
● Completed questionnaire.
● Customer telephone number.

While for the enterprise service *Target range of customer services* is required: a completed questionnaire, telephone contact and offers subscription requirement.

We carry out this step for all modules of enterprise services at the first level. Furthermore, we find links between enterprise services. Based on information from depicted above we can convert first level process model and already draw a diagram of enterprise service collaboration on the first level with the offered and required interfaces.

Now we observe each parent enterprise service from the first level and draw the working composition pattern of daughter component services. We can complete these schemes by mutual interface of daughter services in the particular step of conversion.

RESULTS AND DISCUSSION

SOA and enterprise functionality on business services are interested in information science for at least two decades. Principles of looking at the business process logic and business services logic, according by importance in theoretical logic structures, are not common. We could not find a similar concept of here mentioned approach.

At first, the general role of any logic in any theory is presented in above mentioned approach, and then search for similar roles is depicted in generally procedural logic, process logic and service logic within the enterprise functionality. So we have managed to show that the role of process logic is to describe associations between processes and the context business processes in the surroundings. While on the other hand, the logic of business services forms the broader process units – enterprise services, as separate functional units. This made it possible to move the view of the business functionality through these enterprise services, assign them a high autonomy and distribution. And that is just what SOA brings to the Internet platform 2.0.

There is necessary to mention also using a second aspect of the view. Namely, that showed strong continuity of both logics, i.e., the process logic is the basis for the logic of enterprise services.

SUMMARY AND RECOMMENDATION

Applying SOA changes goals of analysts and programmers to construct the required software enterprise information system. The target software is highly distributed, enterprise services are programmed by using Web 2.0 Internet service. This is the main benefit of SOA due to the new concept of software architecture. Naturally, such architecture based on new flexible architecture units, delivers enterprise management unexpected possibilities in the modification of the target software.

An important problem, however, is the design of enterprise services. SOA paradigm does not bring in this way a satisfactory solution. Many computer scientists using advanced process models, which they converge to primary orchestration and dynamics patterns of services. Surely, we will meet with the publication of constantly more capable of conversion methods, especially for process models according to the method Eriksson – Penker Business Extentions, ARIS method by prof. Scheer and standard BPMN notation.

REFERENCES

ERL, T., 2005: *Service oriented technology: concept, technology and design*. USA: Prentice Hall. ISBN 0-13-185858-0.

WOODS D., MATTERN T., 2006: *Enterprise SOA. Designing IT for Business Innovation*. 1st edition. USA: O'REILLY®. ISBN 0-596-10238-0.

ERL, T., 2006: *Service-oriented Architecture. A Field Guide to Interacting XML and Web Services.* USA: Prentice Hall. ISBN 0-13-142898-5.

MIŠOVIČ, M., 2007: *Application Architectures of Enterprise Information Systems versus Service Oriented Architecture.* Acta univ. agric. et silvic. Mendel. Brun., LV, 6: 233–242. ISSN 1211-8516.

BELL, M., 2008: *Service-oriented Modeling (SOA) Service Analysis, Design, and Architecture*. USA: John Wiley & Sons. ISBN 978.0-470-14111-3.

TOMÁŠŮ, M., 2012: *Modelování podnikové procesní architektury a návrh struktury komponent informačního systému v oblasti služeb*. Diplomová práce. Brno: Mendelova univerzita v Brně, Provozně ekonomická fakulta, Ústav informatiky.

MIŠOVIČ, M., RÁBOVÁ, I., 2012: *Enterprise services and object-oriented methodologies*. Acta univ. agric. et silvic. Mendel. Brun., LX, 7: 179–188. ISSN 1211–8516.

Address

prof. RNDr. Milan Mišovič, CSc., Ing. Jan Turčínek, Department of Informatics, Mendel University in Brno, Zemědělská 1, 613 00 Brno, Czech Republic, e-mail: misovic@mendelu.cz, turcinek@mendelu.cz