Alignment of Business and Application Layers of ArchiMate

Oleg Svatoš[1], Václav Řepa[1]

[1]Department of Information Technologies, Faculty of Informatics and Statistics, University of Economics, Prague
repa@vse.cz, svatoso@vse.cz

Abstract. ArchiMate is widely used standard for modelling enterprise architecture and organizes the model in separate yet interconnected layers. Unfortunately the ArchiMate specification provides us with rather brief detail on the relations between the individual layers and the guidelines how to have them aligned. In this paper we focus on the alignment between business and application layers especially alignment of business processes and application functions, which we see as crucial condition for proper and consistent enterprise architecture. For this alignment we propose an extension for ArchiMate based on the development done in Methodology for Modelling and Analysis of Business Process (MMABP). Application of the proposed extension is illustrated on an example and its benefits discussed.

Keywords: business process model·Enterprise Architecture·ArchiMate·TOGAF·MMABP·functional model·object life cycle

1 Introduction

The enterprise architecture plays an important role in current corporate management. Enterprises realized that after the initial spontaneous growth and accumulation of different legacy systems over the years, it is necessary to create the living organism forming an enterprise under some kind of control in order to allow managing it. The enterprise architecture is one of the tools which help them to do it.

There are different frameworks for capturing the enterprise architectures like FEAF [20], DoDAF [21], ISO 19439 [22], etc. out of which the most popular standard is the TOGAF [14] accompanied by ArchiMate as the modelling language [1]. ArchiMate splits the enterprise architecture into several layers, having for each special elements to capture it with and in TOGAF a method (Architecture Development Method) how to create such layered architecture.

The one of important goals of such architecture framework should be to ensure that the individual architectures are aligned with each other. The specification of ArchiMate is very brief on how the elements of each layer are related and the guidelines, how to make these layers consistent, are very loose. Generally we can say that the specification limits its guidance to the relationships illustrated at Figure 1.

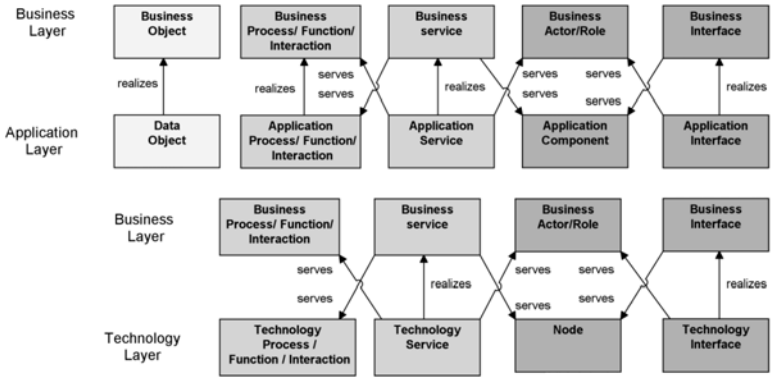

Fig. 1. Relationships between Business Layer and Application and Technology Layer Elements [1] (p.96)

From Figure 1 we can get a "feeling" of how the layers are mutually related. Nevertheless, for complete understanding how these layers are related and what should be the rules for alignment of particular elements from individual layers is this scheme strongly insufficient.

The purpose of this paper is not to elaborate all the issues with alignment of the individual layers there are, as it is too complex for just one reasonably long paper, instead, we focus on one complex issue and that is finding a way how to ensure that the application functions from application layer are aligned with the requirements rising from business processes in business layer. From our point of view this is one of the most important issues that the businesses have to deal with, so that the applications and also the technology assets are optimally supporting the particular business captured in the business layer. In other words, the functions in the application layer should cover all the information transforming and manipulating requirements rising from the business processes in the business layer. The importance and complexity of the business and application layer alignment also rises from the fact that alignment of these layers is dependent on cooperation of different departments throughout the company unlike the alignment of application and technological level which is usually under management of one department (ICT).

In this paper we propose guidelines for the business and application layer alignment, based on advances made in MMABP methodology [2], especially on the introduction of the concept of levels of process abstraction [12] together with well-defined relations among its basic models. The compatibility of the MMABP and ArchiMate was already discussed in [13] and the alignment of the two methods was demonstrated on matching two levels of process abstraction. In this paper we extend the proposed extension to full coverage of the process abstractions and show the benefits rising from it in case of alignment of business and application layer.

The paper is organized in five main sections. After this Introduction section we present the methodology, which the ideas, presented in this paper are based on. We focus mainly on those features of the methodology, which are essential for the pre-

sented ideas. Also the basic features of ArchiMate, relevant to the topic, are briefly presented in this section. In the special section we present the proposal of the extension of ArchiMate oriented on the overcoming of its limitations discussed in the Introduction section. In the following Discussion section we discuss important connected topics including also the structure of the process-driven information system and its relationship to the ERP systems. Conclusions section than summarizes main ideas of the paper, discusses its wider consequences and outlines the needed future work in this field.

2 Methodology

The ideas of proposed extension of the ArchiMate as well as the initial principles which the essential need of such extension comes from, are based on the Methodology for Management and Analysis of Business Processes (MMABP) [2].

2.1 MMABP

MMABP is a methodology for modeling "business systems". By "business system" we generally understand the system of human activities leading to creating the values. MMABP models aim to cover the business system itself as well as its basic infrastructure – information system. The complete MMABP business system model consists of the set of inter-related models of three basic kinds (see Figure 2).

First two kinds of models form the model of the business system itself. Model of the business system consists of the model of being (ontological model) and the model of behavior (intentional model). Being (so-called Real World ontology) is modeled with two basic diagrams from the Unified Modelling Language (UML) [15]:

- Class Diagram represents the system view of being. By this diagram we model objects and their mutual relationships.
- State Chart represents the particular view of being as a temporal model of the single object. By this diagram we model the object's life cycle.

Behavior in the business system is also modeled with two kinds of diagrams:

- Global model of processes represents the system view of intentional behavior: model of business processes and their mutual collaboration.
- Model of the process run represents the particular view of behavior as a temporal model of the single business process.

MMABP defines four levels of abstraction of a process:

a) *Enterprise functionality* is modeled as a system of mutually related enterprise functions, each representing relatively standalone system of business processes.
b) *Process map* represents a system of mutually related business processes of one enterprise function.
c) *Process step* is a part of the single business process representing an internal activity of the process which need not be interrupted by an external event.

d) *Activity* is a part of the process step corresponding to the single change of the state of related business object.

Abstraction levels a) and b) belong to the Global model of processes while levels a) and d) belong to the Model of the processes run. In other words, in MMABP there are two abstraction levels of the system view of processes and two abstraction levels of the temporal view of a process. Each abstraction level is related to the aspect of the business system which MMABP regards as essential: standardization of enterprise functionality (level a)), individualization of enterprise processes (level b)), individual collaboration of processes (level c)), and general causality of the business (level d)).

We can find different approaches to levels of process abstractions in number of different standards and methods ([8]; [23]; [24]; [9]). MMABP has its own synthesizing method build on [12] in which the individual approaches we analyzed and synthesized into four level approach on which MMABP now builds on.

Figure 2 outlines also basic relationships between the ontological (business objects) and intentional (business processes) dimensions of models. Objects from the Class Diagram manifest themselves in the process model as products, inputs, outputs, actors, and other kinds of process aspects. Relationships among objects from the Class Diagram then represent the essential business rules and restrictions (i.e. modality of the business system) which the processes have to respect and fulfill. Looking from the opposite side one can see the business processes from the process diagrams as the ways of fulfilling the modality of the business system in terms of achieving individual business goals. On the detailed level, the life cycles of objects (State Charts) can be seen as definitions of essential causality related to the individual object, while detailed process models (BPMN models) as intentional combinations of the lives of related objects.

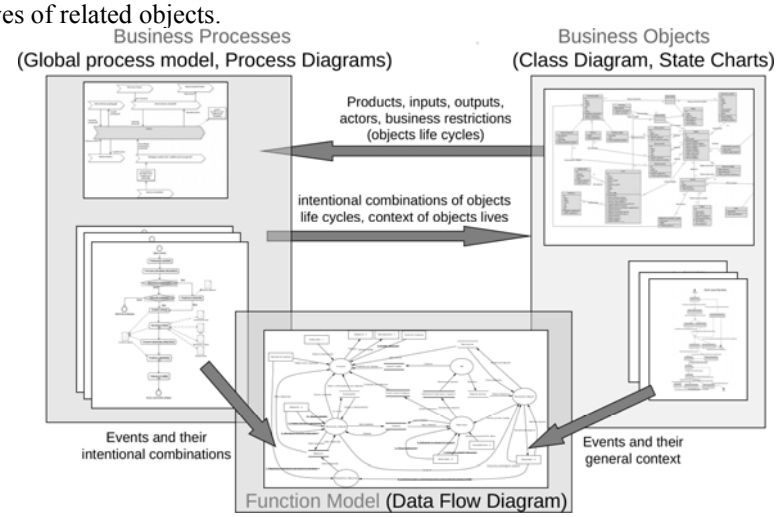

Fig. 2. MMABP models and their essential relationships

The third kind of models is the model of the information system's functionality. This model is represented by the Data Flow Diagram (DFD) as a basic tool for modeling the functionality of information system. DFD has evolved from the so-called Activity

Diagrams used in the SADT methodology [6]. SADT (Structured Analysis and Design Technique) is a methodology, used since about the mid seventies. DFD is the most thoroughly methodically seized in the work of Edward Yourdon [17], [18]. Data Flow Diagram (DFD) plays the crucial role in the Structured Analysis by E.Yourdon [17].

The main reason for using DFD in the MMABP is the fact that DFD and its methodological particulars, like the rules of use, and techniques for linking with other diagrams, represents a great methodology value. Edward Yourdon in [17] introduces the technique for the functional model development based on the analysis of events, so-called Event Partitioning Approach. This technique is very important for linking of DFD with other diagrams in MMABP. It allows MMABP to use events as universal elements which occur in all models at exactly the same level of granularity[1], thus they can be used as a universal meeting point of the contents of all models for tuning all models together in terms of their mutual consistency.

Figure 2 illustrates also basic ways of linking particular models of the business system with the model of the functionality of its information system (DFD) with use of events. Functionality of the information system of the given business system has to be derived from both the causality of the business, and the business goals and ways of their achieving. Business object models (i.e. contextual Class Diagram and related State Charts describing life cycles of selected objects) contain the information about events and their general context in terms of the causality of the business system. Models of business processes complete the information about events and their intentional context; intentionality in the business system.

More details about MMABP particular consistency rules for tuning business system models with DFD can be found in [10].

2.2 Alignment of Function Model and Business Processes Model in MMABP

MMABP considers an activity as the lowest level of process abstraction and it is defined as a unit of work that changes particular state of an important object to another [12]. The focus on states from objects' life cycles is essential as they define how the people of business think about their business and what they find important. This is reflected in life cycles of individual objects, as the life cycle states represent moments in an object life, the business recognizes and therefore it has a name and meaning for it. Activities then represent work done which output has meaning for the business. For instance in process of an order creation there are individual order items added (goods, address, etc.). Operations like add/remove order item usually do not change a state of an order and therefore these operations we would not find in the process model in form of corresponding activities. They would be part of an activity which would be

[1] Each event is defined by the particular contents together with the particular moment of occurrence. So, more events defined as occurring always together, at the same time, have to be regarded as just one single event. This principle alines the granularity of the event to the same level in all possible points of view, which address this event in terms of the same contents.

covering the whole order creation procedure which output state would be the submitted order (see Figure 4). This way an activity may include several operations, but the operations are not the next level of process abstraction as they do not represent intentional behavior any more, only processing functions with inputs and an output. The intentional behavior is hidden in the capabilities of the activity performer (the actor) who, using one's skills and knowledge, combines the operations in such way which fits to the actual situation and keeps the course of processing in the direction of the activity's goal (the associated object state). These operations can be directly linked to functions in a data flow diagram at the highest level of detail, where the functions represent particular transformations, same as the operations do in the life cycle diagrams.

3 ArchiMate

ArchiMate [14] is an enterprise architecture modeling language based on TOGAF framework which is one of the most popular and accepted frameworks for enterprise architecture [14].

There are defined three core layers (business, application, technology) the enterprise architecture is then described in, which can be in full framework extended by additional layers (strategy, physical, implementation and migration). Each core layer consists of active, passive and behavior elements and the behavior elements, on which our analysis focuses. The behavior elements are processes, functions, interactions, events or services. For each layer there are specific definitions of these elements, which then specify the differences among the same concepts in different layers, for instance, differences between business process, application process and a technology process.

For the business layer there is no further detail elaborated in the specification, although, the coexistence with the detailed BPMN models is assumed [14] p.152. When looking at examples in the specification ([14], example 23 on p. 64) one can see that the elaboration of this business process detail is missing in the specification as the business process elements are used at higher level of detail that their definitions originally assume. In some cases they remind more of particular activities than business processes.

In application layer our focus is aimed at the application functions which alignment with the business processes is subject of this paper. An application function is defined as "an application function represents automated behavior that can be performed by an application component" and we assume, judging from its definition, meta-model and the example in [14], p.76, that we can consider the application function as the lowest abstraction of behavior in the application layer. This element then represents behavior of the application which is available to support or even enable the business process execution.

Alignment of the two elements (business processes and application functions) is in the specification solved quite simply. In [14], p.95 it is just stated that the application

(process/function) realizes the business (process/function). No further detail is available nor it is possible to derive from the meta-models provided.

4 Extension of ArchiMate

The presented alignment of information functions and the requirements rising from the detailed business processes in MMABP can be incorporated into ArchiMate framework. The substance of this alignment stands on the defined levels of process abstraction with which we will extend the ArchiMate business layer.

In [13] there was already done the incorporation of process abstraction levels into ArchiMate business layer in case of the top two levels of process abstraction. There were identified two ArchiMate's concepts (business function and business process) that fitted well the definition of the top two levels of process abstraction of MMABP, specified in [12], making so the first two levels of process abstraction of the MMABP and ArchiMate aligned. As the top two levels are aligned and the ArchiMate recognizes the BPMN as standard for detailed business process models [14], p.152, the rules for the two detailed process abstraction levels (process steps and activities) from MMABP can be incorporated into the ArchiMate too. Reader should note that even though we are talking about detailed business processes, the process steps and activities are still reasonable abstract levels for architecture as they do abstract from the implementation. The very detail lies in the operations in the objects' life cycles.

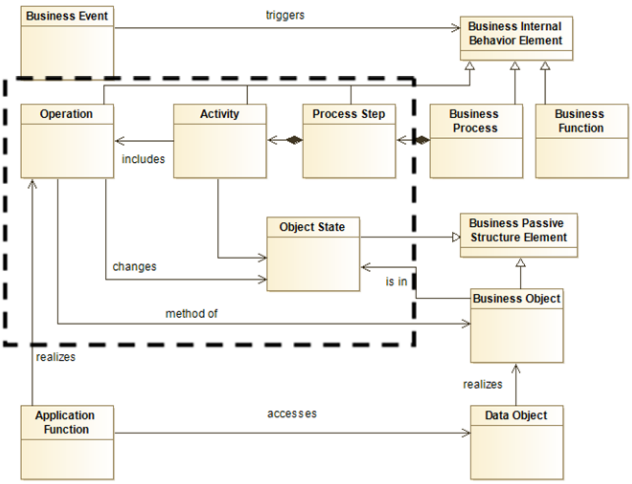

Fig. 3. Relevant fragment of the ArchiMate metamodel with proposed extension (marked area)

As BPMN and its coexistence with ArchiMate models is assumed in the ArchiMate specification, there has to be introduced only one new diagram which is not referenced directly by the ArchiMate specification – the state transition diagram. This diagram is not completely strange for the ArchiMate as the underlying TOGAF works

with a similar type of diagram – the data life cycle diagram [14], p. 103. We propose to use the state machine diagram from UML [15] as ArchiMate has defined the way how to cooperate with this language [13, 19], and to extend the ArchiMate metamodel by the relevant concepts from MMABP as depicted at Figure 3. The original ArchiMate metamodel consists of several fraction models organized according to types of elements, relationships, layers, etc. Figure 3 contains only those elements and their relationships from the ArchiMate metamodel which are relevant to the proposed extension.

The procedure which would enable the business process and application functions alignment evaluation would look the following way.

First, the business processes, alignment of which one is to evaluate, have to be modeled in BPMN at the lowest level of process abstraction - i.e. in our case at the activity level. As an activity is bound to particular state from an object life cycle, one has to model the relevant object life cycles in parallel. There should be taken into consideration that the associations of a business object itself, which we can find in passive structure model, should be reflected in operations in the business object life cycle covering the possible adding and removing of these associations to other business objects. In the operations specified in object life cycle diagrams we are then getting business specification of those application functions which support the process execution. Then we can either try to match the specified operations with existing application architecture and the relevant application functions and evaluate their alignment or they may serve as specification of business requirements for further application design and implementation. It is important to note, that we are still at architectural level as the application functions do not represent design as recommended [14], p. 73, they just represent reflection of business requirements on what information transforming functions the information system should provide in order to fully support the business process execution.

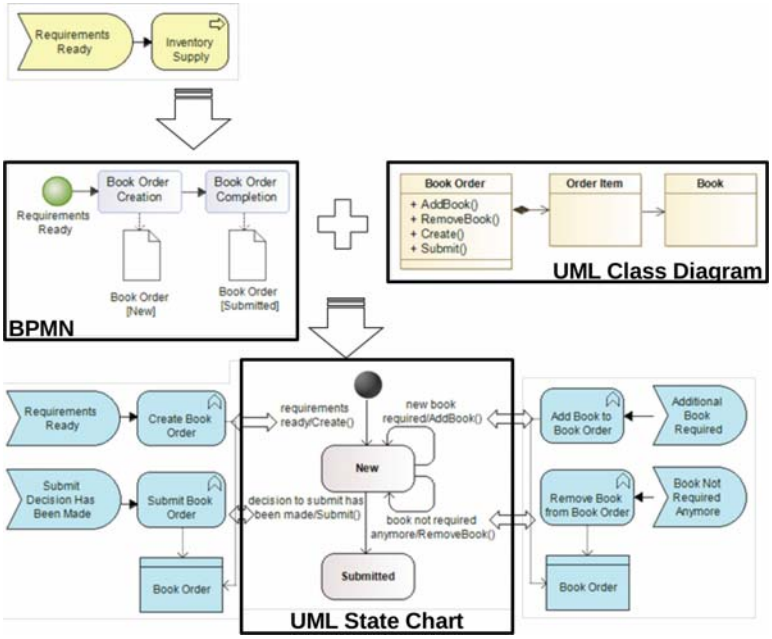

Fig. 4. ArchiMate alignment evaluation procedure example

We illustrate the individual steps from the specified procedure on simple example at Figure 4, which for purpose of this paper, works only with parts of the models concerned, for sake of the clarity of the illustration. Symbols in the figure follow the Archimate color key (yellow elements of business layer, blue elements of application layer). Non-Archimate diagrams (Class Diagram and State Chart from UML, and the BPMN diagram) are labeled with their names directly in the figure.

5 Discussion

The proposed extension extends the meta-model of the ArchiMate as depicted in Figure 3. When compared with the original alignment scheme depicted in Figure 1 we can conclude that the proposed extension makes the alignment of the business and application layer more tangible. The idea behind is to focus on the issue that, from our point of view, matters the most (alignment of business processes and the application functions) and not to dissolve in a creation process of a method which tries to cover all possible relations between all elements of the two layers. When applied, one can evaluate with reasonable certainty whether the two layers are aligned especially whether the functionality of applications fits the requirements of the business and its processes. Without this extension it is hard to do, if not impossible at all, as there is missing relevant detail in the business layer we could match the particular application functions with.

As noted above the reader should always bear in mind that the proposed extension is not covering all the possible issues rising from business and application layer alignment. Its main focus is on alignment of the two core behavior elements – business processes and application functions. Alignment of other behavior, passive and active structure elements is out of the scope of this paper.

Alignment of business functions and processes with the functionality of an information system is very important also from the point of view of the process – driven management theory itself [3],[4]. Designing the system of processes, one has to carefully decide about which key processes the process system consists of and which processes should support them. The difference between key and support processes is essential from different perspectives, including also the perspective of the process dynamics. The dynamics of the enterprise behavior should be concentrated in key processes while support processes represent rather the stable enterprise functionality, needed as support of the performance of key processes. The main meaning of key processes is managerial: their task is to keep the context of particular services by support processes in terms of the final value for the customer. Therefore, key processes are usually called "end to end" processes as they have to cover the whole context of the given business case – delivering the value to the customer from the very beginning (i.e. identification of the customer's need) to the very end. It is obvious that the basic particular services, which the key process consists of, are relatively stable, standard, they are given mainly by the ontological substance of the business. On the other hand, their combinations, in terms of solving particular needs of the individual customer, is always individual. It depends on the given situation, special needs of the given customer, possibilities of technology, and other circumstances, which all put the special demands not only on the instance of the process but also on the development of its general definition. From the point of view of the supporting information system, key processes thus require the dynamic support from the IS while support processes can be covered by relatively stable, standard IS functionality.

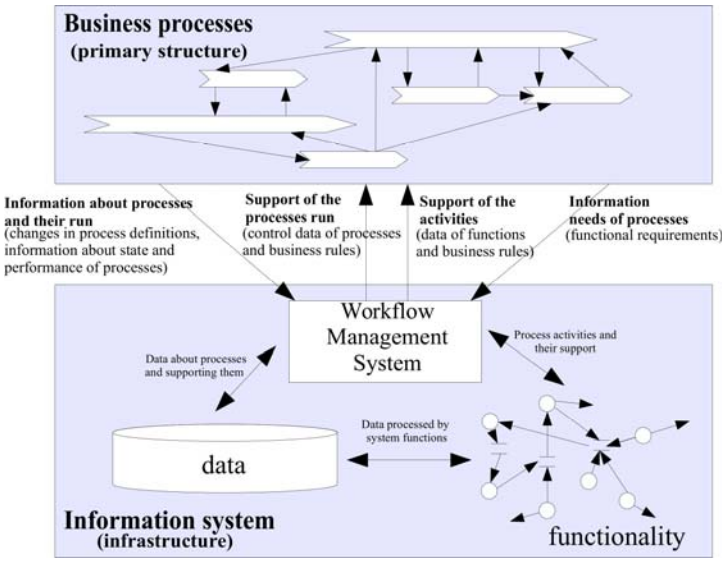

Fig. 5. Process-driven information system according to MMABP

Figure 5 outlines the MMABP idea of the general structure of the information system, based primarily on the model of business processes. According to the idea of process-driven organization [3],[4] the information system has to be able to accommodate its functionality to the permanently changing needs of business processes. This idea directly contradicts with the traditional monolithic conception of information system (IS) in terms of the Enterprise Resource Planning (ERP) which defines the functionality of IS strictly in a static manner. This contradiction is discussed in literature from different points of view ([11], [5] for instance). Nevertheless, making the IS flexible in terms of the process-driven conception does not require complete leaving of the ERP idea of IS functionality. According to MMABP, required flexibility of the IS rests rather in the changes of combinations of essential IS functions in order to meet the changes of business processes, than in changes of the essential functions themselves. Process-driven IS then should consist of some stable parts complemented with the software component which dynamically combines these stable components according to the momentary needs of business processes. Stable parts of the IS represent relatively stable aspects of the business: the database and the standard essential functions. The software component, which ensures the needed dynamics, is based on the business processes definitions and works with the information about the current state of running processes. This allows it to combine the standard essential functions, working with the standard data structures dynamically according to the momentary needs of business processes. Such software component is usually called Workflow Management System (WfMS). WfMS is a regular field of the professional interest including even the standardization since mid-nineties [16]. Such way created IS is able to support the needs of business processes dynamically by serving the standard functionality on standardized data structures based on the information from

definitions of business processes and about their states. This fundamental division of the relatively stable components of the IS from the business processes driven "behavior" of information system allows creation of the fully dynamic IS keeping the maximum advantage of the standard ERP systems at the same time.

6 Conclusions

Alignment of individual layers of ArchiMate model is an important issue that needs to be treated carefully. In this paper we focus on the issue that, from our point of view, matters the most: the alignment of business processes and the application functions. Since ArchiMate is very brief on this topic and does not go into such detail in its specification, which would allow us to specify clear and unambiguous rules for the alignment, we extend the ArchiMate's metamodel and guidelines according to how this alignment is set up in Methodology for Modelling and Analysis of Business Process (MMABP) and utilize the standards the ArchiMate is opened to. The extension includes incorporation of detailed process models in BPMN, object life cycles captured in UML state machine diagram and a procedure how the alignment should be step by step done. The additional effort connected with creation of the newly introduced models, which cover the missing levels of detail in the business layer, rewards the analysts with possibility of proper evaluation of alignment of business and application layer, especially how are the requirements rising from the business processes supported by the application functionality.

In this paper we have focused on the alignment of business process and application functions, as we saw it as core issue that needs to be addressed, but the briefness of the ArchiMate specification on the alignment of individual layers leaves other blank spaces, which currently have to be solved intuitively, and call for further elaboration and solution propositions.

In the future work in this field we plan to focus on the above mentioned alignment of other layers as a further step to the deeper integration of the Enterprise Architecture with the ideas of process-driven management and related methodologies, particularly represented by MMABP.

Acknowledgment

The paper was processed with financial contribution of long term institutional support of research activities by Faculty of Informatics and Statistics, University of Economics, Prague.

7 References

1. The Open Group: ArchiMate 3.0 Specification. Van Haren. 2016. ISBN 94-01-80047-2
2. Business System Modeling Specification: http://opensoul.panrepa.org/meta-model.html (2000-2016).

3. Hammer, M., Champy, J.: Reengineering the Corporation: A Manifesto for Business Revolution. London: Nicholas Brealey Publishing (1993).
4. Hammer, M.: The Process Audit. Harvard Business Review, 85(4), pp. 111 – 123, 2007.
5. Holmberg, N., & Johansson, B.: A Service Oriented Perspective of Enterprise Resource Planning. In Systems. Journal of Systems Integration, 8(2), 2017.
6. Marca, D., McGowan, C.: Structured Analysis and Design Technique, McGraw-Hill, 1987, ISBN 0-07-040235-3
7. Object Management Group: Business Process Model and Notation (BPMN) Specification Version 2.0.2., http://www.omg.org/spec/BPMN/
8. von Rosing, M., von Scheel, H., Scheer, A.W.: The Complete Business Process Handbook: Body of Knowledge from Process Modeling to BPM. Elsevier Science, San Francisco 2015.
9. Rosenberg, A.: SAP Modeling Handbook - Modeling Standards. https://wiki.scn.sap.com/wiki/display/ModHandbook/
10. Řepa, V.: Object-Oriented Analysis with Data Flow Diagram. In Information Systems Development (pp. 419-430), 2013, Springer, New York. ISBN 978-1-4614-4950-8.
11. Srivardhanaa T,, Pawlowski, S.D.: ERP systems as an enabler of sustained business process innovation: A knowledge-based view. In the Journal of Strategic Information Systems, Volume 16, Issue 1, March 2007, Pages 51-69.
12. Svatoš, O., Řepa, V.: Working with Process Abstraction Levels. In: Perspectives in Business Informatics Research. Praha, Switzerland : Springer, 65–79. 2016. ISBN 978-3-319-45320-0.
13. Svatoš, O.: Business Process Modeling Method for Archimate. In: IDIMT-2017 Digitalization in Management, Society and Economy [online]. Poděbrady, 06.09.2017 – 08.09.2017. Linz : Trauner Verlag Universität, 2017, s. 325–332. ISBN 978-3-99062-119-6.
14. The Open Group: TOGAF Version 9.1. Van Haren. 2011. ISBN 90-87536-79-8.
15. UML Superstructure Specification, v2.0 document 05-07-04, Object Management Group, 2004.
16. Workflow Management Coalition, The Workflow Reference Model, WFMC-TC-1003, 1995 (http://www.wfmc.org/standards/docs/tc003v11.pdf).
17. Yourdon, E.: Modern Structured Analysis, Prentice Hall, Englewood Cliffs, N.J., 1989.
18. Yourdon, E.: Just Enough Structured Analysis - rev. 051406 [Online], 2006. Available at http://zimmer.csufresno.edu/~sasanr/Teaching-Material/SAD/JESA.pdf [Accessed 10. 04. 2018].
19. The Open Group: Using the ArchiMate Language with UML, White Paper (W134), www.opengroup.org/bookstore/catalog/w134.htm (2013).
20. Federal Enterprise Architecture Framework Version 2 (2013)
21. The DoDAF Architecture Framework Version 2.02, http://dodcio.defense.gov/Library/DoD-Architecture-Framework/ (2010)
22. ISO 19439:2006, https://www.iso.org/standard/33833.html
23. Process Classification Framework | APQC, https://www.apqc.org/pcf
24. The Supply Chain Operations Reference model (SCOR), http://www.apics.org/sites/apics-supply-chain-council/frameworks/scor