A Method and Tool for Business-IT Alignment in Enterprise Architecture

Alain Wegmann, Pavel Balabko, Lam-Son Lê, Gil Regev, and Irina Rychkova

École Polytechnique Fédérale de Lausanne (EPFL), School of Communication and Computer Science CH-1015 Lausanne, Switzerland
{ Alain.Wegmann, Pavel.Balabko, Gil.Regev, LamSon.Le, Irina.Rychkova}@epfl.ch

Abstract. Enterprise architects seek to align enterprise processes and structure with their supporting IT systems so that enterprises can flourish in their environment. The enterprise architecture (EA) discipline has emerged from business best practices. EA frameworks are therefore mostly informal. As a result there is a lack of EA tools that can help enterprise architects to check this alignment. Most notably, current EA tools do not help enterprise architects to formalize the alignment of the multiple levels that constitute an enterprise model. In this paper we propose an EA framework and an associated tool that provide alignment checking along the functional and organizational hierarchies. We illustrate this approach with the Sun Microsystems industrial example of the Pet Store.

1 Introduction

The design of business and IT system alignment is the domain of Enterprise Architecture (EA) [7]. Enterprise architects seek to align enterprise processes and structure with their supporting IT systems.

Several EA frameworks have been defined in recent years [17]. EA frameworks are characterized by a modeling paradigm where the multi-disciplinary nature of the enterprise and its IT systems are modeled in multiple levels. Each level represents the reality as viewed by a different specialist, e.g. marketing, business process, and IT developers. An acute problem with this modeling paradigm is to verify the alignment and the traceability between the levels.

In this document we represent a solution to this problem in the form of the SEAM EA framework [18] and its accompanying tool SeamCAD [11]. SEAM is grounded in General System Thinking (GST) [19], and Living Systems Theory [12] that explain how to interpret the enterprise reality. The need for reasoning about behavior and construction is a general systems thinking principle that recognizes that system behavior results in a certain construction (usually called structure) which in turn supports future behavior [19], [4].

The two main SEAM concepts used to express the behavior and construction are the functional and organizational levels. The *functional levels* represent the

Proceedings of the CAiSE'05 Forum - O. Belo, J. Eder, J. Falcão e Cunha, O. Pastor (Eds.)
© Faculdade de Engenharia da Universidade do Porto, Portugal 2005 - ISBN 972-752-078-2

behavioral hierarchy whereas the *organizational levels* represent the constructional hierarchy.

SEAM is based on the terminology defined in the Reference Model for Open Distributed Processing (RM-ODP) and uses formal methods (e.g Alloy [13] and ASM [1]) to reason rigorously about the interpretation made of the reality.

We illustrate our solution with the case study of the Pet store [2] released by Sun Microsystems as their first Java Blueprint application. We add to the original Sun Microsystems case study the business system and business process levels in order to demonstrate the alignment between business and IT.

This paper has the following structure: Section 2 defines the concepts of organizational and functional levels; Section 3 illustrates our contribution with the Pet Store example; Section 4 describes the related work. Section 5 presents our conclusions and future work.

2 Organizational and Functional Levels

In this section we describe the concepts of organizational and functional levels and formalize their alignment.

Each organizational level is interpreted as being made up of computational objects that represent systems. To every computational object one can apply two viewpoints [15]: the information viewpoint (IV) and the computational viewpoint (CV). The information viewpoint describes the system seen as a whole (black box specification). The computational viewpoint defines the system seen as a composite (white box specification).

In SEAM a computational object's computational viewpoint (CV) is composed of several component computational objects. These components participate in collaborations. A collaboration (also called joint action [6]) is an action that involves multiple participants. Refining a computational object's CV results in a hierarchy of CVs where each CV represents a model of the system at a specific organizational level.

Mutatis mutandis, each functional level represents behavior at a given level of detail of a computational object. The computational object's information viewpoint defines the behavior with information objects, localized actions, and relations between them. More detailed IV can be obtained by adding new information about system behavior. Refining of an information viewpoint results in a hierarchy of IV's where each IV represents a model of the system at the specific functional level.

The above concepts are formalized in [10], using Alloy 2.0. Alloy is a specification language based on set theory [13]. With this formalization, we were able to build a CAD tool called SeamCAD for modeling hierarchical systems [11].

The SEAM hierarchical model of a system can be considered as a set of partial models where each partial model represents the information viewpoint(s) of the relevant system (or collaboration of its subsystems) of interest at a particular organizational and functional level.

The following definition of functional levels alignment is adapted from [8]:

At a given organizational level o, let $IV_{s,o,f}$ be a partial model representing system s at functional level f and $IV_{s,o,f+1}$ is an extension of $IV_{s,o,f}$ on the functional level f+1. Partial model $IV_{s,o,f+1}$ is said to be aligned with the partial model $IV_{s,o,f}$ if and only if all the behavior described by $IV_{s,o,f}$ is included in the behavior described by $IV_{s,o,f+1}$.

This definition is based on the intuitive meaning of the functional refinement technique: model $IV_{s,o,f+1}$ extends or implements the behavior of model $IV_{s,o,f}$. It implies the essential property of *behavioral substitutability*[16]. In a similar manner we define the alignment for the organizational levels:

For any functional levels $f1 \leq f2$, let $IV_{s,o,f1}$ be a partial model representing system s at organizational level o and $CV_{s,o+1,f2} = IV_{collaboration,o+1,f2} + \sum_i IV_{ss_i,o+1,f2}$ is a partial model representing the collaboration of subsystems ss_i of the system s on the next organizational level o+1. Partial model $IV_{s,o,f1}$ is said to be aligned with the partial model $CV_{s,o+1,f2}$ if and only if all the behavior described by $IV_{s,o,f1}$ is included in the behavior described by $CV_{s,o+1,f2}$ and mapping of vocabulary between organizational levels is defined and complete.

Vocabulary mapping is needed to relate concepts between organizational levels. For example, the meaning of an *Order* information object can be different for the company itself, where *Order* represents a purchase, and for the sales department where *Order* is a collection of customer's details and product specification.

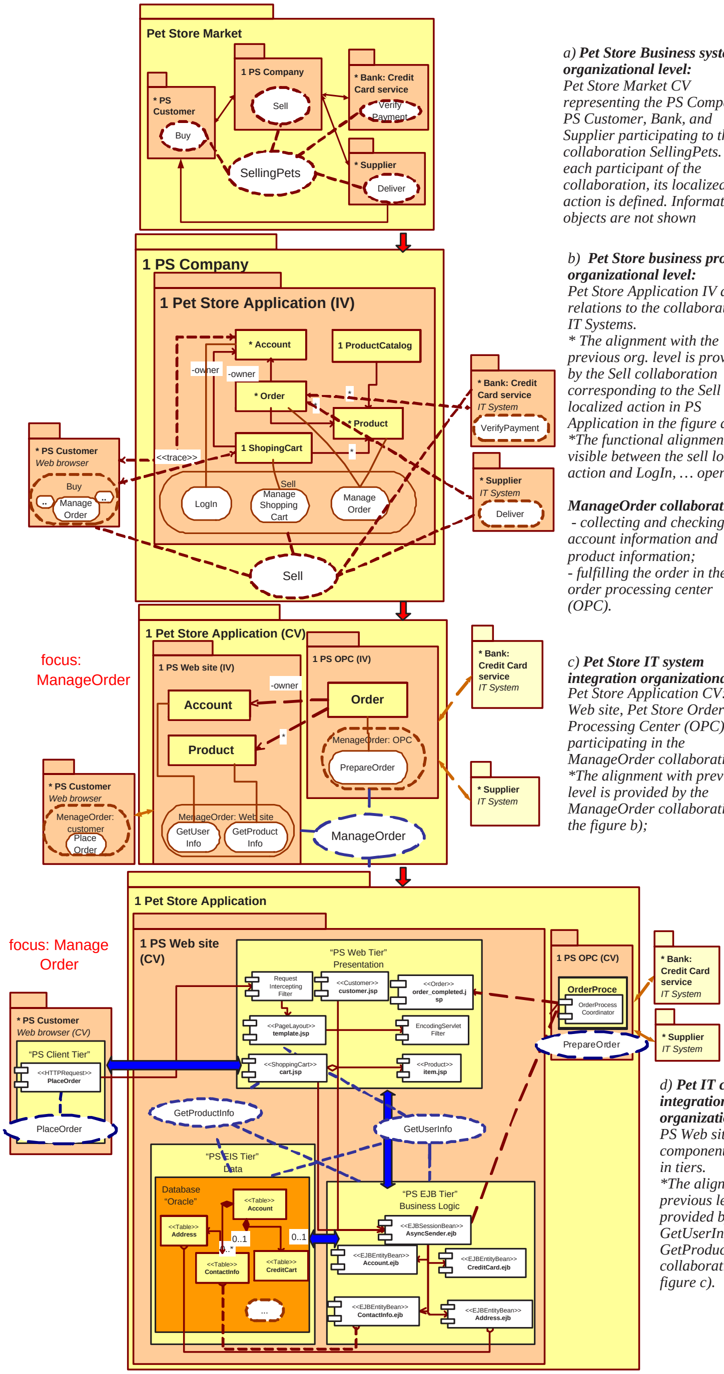
The alignment checking of the SEAM graphical models is based on The Abstract State Machine (ASM) paradigm and tools such as the AsmL language and the Asmlt test tool developed by Microsoft research group [1], [9]. This is performed by exporting the SEAM graphical model as XML and then translation the XML file into the ASM specification (asml model). The resulting asml model can be simulated using the Asml environment, and tested using Asmlt. During conformance testing, Asmlt produces the trace of state transitions of the two models to compare, and reports errors. The errors show the behavioral nonequivalence between two models that means the violation of the behavioral substitutability property.

3 Modeling Example: The Sun Mycrosystem's Java Pet Store

In this section we illustrate the SEAM framework with the example of the Pet Store [2]. The Java Pet Store application is a realization of a distributed internet - shop, selling pets to its customers.

Figure 1 represents four organizational levels. In some of these organizational levels, multiple functional levels are represented simultaneously.

a) The Pet Store business system organizational level (Figure 1.a). This level represents the Pet Store Market, where the Pet Store Company (owner of the Pet Store Application) is shown in collaboration with other business entities, such as Bank, Pet Store Customer, and Supplier. The Pet Store Market is not defined in the original Java Pet Store model. However, in SEAM we con-



*a) **Pet Store Business system organizational level:*** *Pet Store Market CV representing the PS Company, PS Customer, Bank, and Supplier participating to the collaboration SellingPets. For each participant of the collaboration, its localized action is defined. Information objects are not shown*

*b) **Pet Store business process organizational level:*** *Pet Store Application IV and relations to the collaborating IT Systems.*
** The alignment with the previous org. level is provided by the Sell collaboration corresponding to the Sell localized action in PS Application in the figure a) ;*
**The functional alignment is visible between the sell loc action and LogIn, … operations*

ManageOrder collaboration:
- collecting and checking user account information and product information;
- fulfilling the order in the order processing center (OPC).

*c) **Pet Store IT system integration organizational level:*** *Pet Store Application CV: PS Web site, Pet Store Order Processing Center (OPC) participating in the ManageOrder collaboration.*
**The alignment with previous level is provided by the ManageOrder collaboration in the figure b);*

*d) **Pet IT components integration organizational level*** *PS Web site CV: J2EE components organized in tiers.*
**The alignment with previous level is provided by the GetUserInfo and GetProductInfo collaborations in the figure c).*

Fig. 1. Pet Store SEAM hierarchical model. Organizational and functional alignment

sider it important to describe the business context in which the system of interest operates.

b) The Pet Store business process organizational level (Figure 1.b). This level represents the Pet Store application with its information viewpoint in the context of the Pet Store Company. The Pet Store Market is hidden. Necessary information concepts were added to the model to reflect the Pet Store Application functionality on the particular abstraction level. Note that two functional levels are represented simultaneously in this model.

The Pet Store Customer Web browser, Bank and Supplier IT Systems in this model represent business partners from the business organizational level. For simplicity their contexts were omitted as they are not our systems of interest.

c) The Pet Store IT system integration organizational level (Figure 1.c). On this organizational level the Pet Store application can be observed within its computational viewpoint as a collaboration of the Pet Store Web Site and OPC (Order Processing Center).

The IV of a system of interest defined in the previous organizational level makes explicit the system responsibilities. In the current organizational level we distribute these responsibilities between the relevant subsystems. In our example the Pet Store Application IV defined on the figure 1.b is distributed between Pet Store Web site and Pet Store OPC on the figure 1.c. This is the essence of the SEAM design process.

Note that as in the previous organizational level, two functional levels are represented simultaneously in this model.

d) The Pet Store IT components integration organizational level (Figure 1.d). This level describes the construction of the Pet Store application multi-tiered organization as defined in [2]. The model illustrates the ManageOrder part of the Pet Store behavior implemented with J2EE components.

The Pet Store example highlights the key features of the SEAM framework: the explicit modeling of systems across organizational and functional levels; the unified graphical notation; explicit relations between models on different levels; functionality-to-organization mapping; model alignment within one and between different organizational levels.

4 Related Work

Many "empirical" EA methods exist, e.g. FEA, TOGAF. For an exhaustive list see [17]. In Information Systems, methods such as DEMO [3] and OPM [5] and in Requirements Engineering, methods such as TROPOS [14] have been proposed to design IT systems aligned with business requirements. All these methods provide traceability by top-down design. SEAM, on the other hand, makes the relations between what is represented in the organizational and functional levels explicit and defines the concept of organizational and functional alignment using the principle of behavioral substitutability.

5 Conclusion

In this paper we described the need for having functional and organizational levels in EA frameworks to reason about business and IT alignment. We proposed the concepts of organizational and functional alignment which is based on the behavioral substitutability principle. We represented the SEAM EA framework and its associated tool SeamCAD that provide both hierarchies and formalizes their alignment. We illustrated our method with the example of the Pet Store.

References

1. Börger, E., Strk, R.: Abstract State Machines: A Method for High-Level System Design and Analysis. Springer-Verlag, 2003.
2. Designing Enterprise Applications with the J2EE Platform, Second Edition `http://java.sun.com/blueprints/guidelines/`
3. Dietz, J. L. G.: DEMO: towards a discipline of Organisation Engineering. 1999.
4. Dietz, J.: Basic notions regarding business processes and supporting information systems, Proceedings of the 5th BPMDS Workshop, Riga, Latvia, June 2004.
5. Dori, D., Object-Process Methodology, A Holistic Systems Paradigm. 2002: Springer Verlag.
6. D'souza, F.D., Wills, C. A.: Object, Components and Frameworks with UML, The Catalysis Approach. 1999: Addison-Wesley.
7. Duffy, J. IT/Business Alignment: Delivering Results, CIO Magazine, 2001, retrieved from `http://www.cio.com/analyst/123101_idc.html` on September 2004.
8. Edwards, S. H., Gibson, D. S., Weide, B. W., Zhupanov, S.: Software component relationships. In Proceedings of the Eighth Annual Workshop on Software Reuse, March, 1997
9. `http://research.microsoft.com/fse/asml/`
10. Lê, L.S., Wegmann, A.: Definition of an Object-Oriented Modeling Language for Enterprise Architecture. System Sciences, 2005. HICSS '05. Proceedings of the 38th Annual Hawaii International Conference on , 03-06 Jan. 2005 Pages:222a - 222a
11. Lê, L.S., Wegmann, A.: SeamCAD 1.x: User's Guide, School of Computer and Communication Sciences, EPFL, Lausanne Switzerland, Report No. IC/2004/98, November 2004.
12. Miller, J.G.: Living Systems. University of Colorado Press, 1995.
13. MIT, The Alloy Constraint Analyzer, `http://alloy.mit.edu/`
14. Mylopoulos, J., Kolp, M., and Castro, J. "UML for Agent-Oriented Software Development: The Tropos Proposal," Proc. UML 2001.
15. OMG, ISO/IEC 10746-1, 2, 3, 4 — ITU-T Recommendation, X.901, X.902, X.903, X.904, Reference Model of Open Distributed Processing. 1995-1996.
16. Philippi, S.: Formally based modeling and inheritance of behaviour in object-oriented systems. Journal of Systems and Software, Feb 2004.
17. Schekkerman, J. How to Survive in the Jungle of Enterprise Architecture Frameworks: Creating or Choosing an Enterprise Architecture Framework, Trafford, Canada, 2003.
18. Wegmann, A.: On the systemic enterprise architecture methodology (SEAM). In: Proceedings International Conference on Enterprise Information Systems (ICEIS 2003), Angers, France.
19. Weinberg, G. M.: An Introduction to General Systems Thinking. New York: Wiley & Sons, 1975.