Computer Standards & Interfaces 35 (2013) 277–293

Contents lists available at SciVerse ScienceDirect

Computer Standards & Interfaces

journal homepage: www.elsevier.com/locate/csi

Hierarchy-oriented modeling of enterprise architecture using reference-model of open distributed processing

Lam-Son Lê [a,*], Alain Wegmann [b]

[a] School of Computer Science and Software Engineering, Faculty of Informatics, University of Wollongong, New South Wales 2522, Australia
[b] School of Computer and Communication Sciences, École Polytechnique Fédérale de Lausanne (EPFL), Lausanne 1015, Switzerland

ARTICLE INFO

Available online 16 February 2012

Keywords:
Enterprise architecture
Enterprise modeling
Systems theory
RM-ODP
SEAM
Alloy

ABSTRACT

Modeling Enterprise Architecture (EA) requires the representation of multiple views for an enterprise. This could be done by a team of stakeholders that essentially have different backgrounds. One way to do this is to structure the model into hierarchical levels each of which can be of interest to just some, not all, stakeholders. Due to the multidisciplinary nature of EA, stakeholders simply cannot choose a single modeling approach, even a widely-recognized one, to build their enterprise model and make it viewable and understandable to the whole team. Developing a modeling framework that can be applied uniformly throughout the entire enterprise model and that can be used by all stakeholders is thus challenging. We based our work on the RM-ODP (Reference Model of Open Distributed Processing) – a standardization effort that defines essential concepts for modeling distributed systems, as well as ODP-related international standards/recommendations, to develop such a modeling framework that we call SeamCAD. This framework consists of a computer-aided tool and a language behind the tool for modeling EA in a hierarchical manner. SeamCAD makes RM-ODP applicable in the context of multi-level EA and consolidates the SEAM – a family of methods for seamless integration between disciplines.

1. Introduction

Enterprise Architecture (EA) captures the whole vision of an enterprise in various aspects regarding both business and information technology (IT) resources [47]. In EA, the goal is to align the business resources and IT resources to maintain or improve the competitiveness of the enterprise. EA is a discipline that analyzes the services offered by an enterprise and its partners to the customer, the services offered by the enterprise to its partners and the organization of the enterprise itself and of its IT. Making an EA project can, for example, help the enterprise gain more customers, reduce the operation costs or increase its agility. This can be done by better identifying the services that the enterprise provides to the customer, by removing the duplication and inconsistencies in business processes and/or information flow, by giving the management more IT-supported facts for facilitating decision making.

During an EA project, an EA team – typically a multidisciplinary team – develops an enterprise model that represents the enterprise, its environment and its internals. The representation of the enterprise can include various aspects such as the services offered by the enterprise, the IT systems, as well as their implementation in terms of business processes and IT applications. Working with a model is important. When making the model, the team develops an agreed and shared representation of the enterprise, its environment and its internals. They also define what the project needs to achieve.

Being motivated by the challenges of defining a modeling technique that can be applied uniformly across hierarchical levels of an enterprise model, we developed a toolkit and eventually came up with a language definition for modeling EA in a hierarchy-oriented manner. We called this framework SeamCAD. This work was part of (and actually consolidates) an EA methodology called Systemic Enterprise Architecture Methodology (SEAM) that has an established pedigree in the literature and consulting services. The underlying rationale of SeamCAD was originally established in SEAM [57,54], which is based on the Reference Model of Open Distributed Processing (RM-ODP) – a joint effort by ISO[1]/IEC[2] and ITU-T,[3] which provides a co-ordinating framework for the standardization of open distributed processing [23]. To be able to make diagrammatic representations capturing different aspects of EA and to put them together in a coherent model that is manageable in a computer-aided tool, we i) further refine the SEAM building blocks using viewpoint-specific modeling concepts of the RM-ODP; ii) formally define the well-formedness of enterprise models; iii) explicitly describe the traceability between model elements of an enterprise model by leveraging the viewpoint correspondences defined in RM-

* Corresponding author.
E-mail address: lelamson@gmail.com (L.-S. Lê).

[1] International Organization for Standardization.
[2] International Electrotechnical Commission.
[3] International Telecommunication Union.

0920-5489/$ – see front matter
doi:10.1016/j.csi.2012.01.008

ODP. To make the diagrammatic representations more precise, we annotate them with declarative specifications written in a formal language. We had a few applications of SeamCAD both in the industry and within the academic setting. We invited EA practitioners and our master's students to test the framework.

The rest of this article is structured as follows. Section 2 outlines the key principles of the SEAM method. Section 3 discusses the motivation for our work on modeling EA. Section 4 proposes how the concepts defined in Part 2 and Part 3 of RM-ODP can be combined to enrich EA models, especially on the aspects of business functions and information processing. Section 5 comes up with a meta-model and provides some insight into the SeamCAD toolkit. Section 6 presents the applicability of SeamCAD and the user's feedback we obtained from external practitioners and our master's students. Section 7 surveys related work. Section 8 concludes the article and points out our future work.

2. Systemic Enterprise Architecture Methodology (SEAM) and its RM-ODP foundations

SEAM is a family of methods for seamless integration between disciplines. SEAM for Enterprise Architecture is an enterprise architecture method that belongs to the change management category [59]. To simplify the discourse, we will use the term SEAM to designate SEAM for Enterprise Architecture in the remainder of this article. Enterprise architects can use SEAM, to develop an enterprise model, a model that represents the relevant features of the organization and its environment. These features depend on the nature of the project for which the model is necessary. They may span the markets in which the organization operates and the implementation of the IT systems that support its operations.

One of the key principles of SEAM is that EA modeling should be done in a systematic way across all hierarchical levels that are created by perceiving enterprises as hierarchical [52]. Note that this perception does not necessarily mean that enterprises are essentially hierarchical. However, we believe that modeling EA in this way has an advantage because people tend to reason in terms of hierarchy [7]. SEAM has another essential principle that is about how enterprise entities are interpreted and represented in an enterprise model. In SEAM, all entities are systematically treated either *as a whole* or *as a composite*, depending on the view [57].

SEAM takes its foundations from the RM-ODP. This standard defines a modeling infrastructure for complex distributed IT systems. RM-ODP is composed of four parts [23]. Part 1 is an overview of RM-ODP and is non-normative. Part 2 defines the fundamental concepts needed for modeling Open Distributed Processing systems. Part 3 presents an application of Part 2 for particular viewpoint specification languages (i.e. enterprise, information, computational, engineering, technology viewpoints). Part 4 is a partial formalization of the previous parts.

SEAM does not rely on the RM-ODP viewpoint specification languages but seek to extend basic modeling concepts defined in Part 2 in order to represent systems that span business and IT systematically [53,30]. The rationale behind this systemic principle was to uniformly apply the same modeling techniques regardless of the subject to be modeled (e.g. business or IT systems) and to have a relatively small set of heuristics for specific aspects of each subject [57].

RM-ODP Part 2 defines the terms *abstraction* and *atomicity*. It is specified that *fixing a given level of abstraction may involve identifying which elements are atomic*. SEAM has two different kinds of levels of abstraction: functional and organizational. In the functional hierarchy, the element that determines the level of abstraction is the action. In the organizational hierarchy, it is the object the modeler considers as a whole that determines the level of abstraction.

3. Motivation

3.1. Example

Let us consider an example of a bookstore whose management decides to provide the company's services via the Internet. The management has a goal to specify the services that the bookstore can provide its customers with and to describe how to implement them using business and IT resources. A book-selling market contains a `BookValueNetwork` and a `Customer`. The value network consists of three companies: a bookstore company named `BookCo` (responsible for the service of processing the orders placed by the customer), a shipping company called `ShipCo` (responsible for shipping the books ordered) and a publishing company `PubCo` (responsible for supplying the books that were ordered but not yet available in the inventory of the bookstore company). The departmental structure of the bookstore company shows two departments: one for coping with the purchasing data (`PurchasingDep`) and the other for managing an inventory of books (`WarehouseDep`). We might have an additional level showing the IT infrastructure of these departments.

Fig. 1 gives a simplified representation of the organizational structure and services in the bookstore enterprise context using an ad-hoc notation. A regular rectangle represents a business entity or an IT system. A rounded rectangle can be attached to a regular rectangle to represent the main service offered by the business entity or the IT system drawn under the regular rectangle. The smile symbol stands for people. The lines connecting these entities and people denote the containment hierarchy. In this project, the EA team needs to model the business entities, the IT systems (drawn under regular rectangles in Fig. 1) and their environment, the services offered to the customer by these entities, the company to company (and department to department) business processes, information flow and interaction between the IT system and a clerk who operates it and possibly the overall architecture of the IT system.

3.2. Challenges in modeling EA hierarchically

As exemplified in the Bookstore example presented in Subsection 1, modeling EA involves presenting multiple views of an enterprise that show multiples business entities, IT systems and the services they offer. One way to do this is to structure the model into hierarchical levels (e.g. market level, value network level, company level) each of which can be of interest of just some, not all, stakeholders. Due to differences in their background, the stakeholders may not want to use a single common modeling approach, even a widely-recognized one, to build the enterprise model.

Developing a modeling framework that can be applied uniformly throughout the entire enterprise model and that can be used by all stakeholders is challenging. Firstly, the framework should have a uniform approach for specifying the services (e.g. `mall Selling book`, `Processing order`) offered by business entities and IT systems and for describing their implementation across hierarchical levels. Secondly, the framework should allow the stakeholders to represent the service specification and the service implementation of multiple business entities and IT systems, even within the same hierarchical level (e.g. both `BookCo` and `ShipCo` are to be represented in detail). Thirdly, the services offered by those entities and systems should be expressed at different levels of granularity (e.g. `Selling book` is broken down into `Getting order`, `Payment` and `Delivery`). Fourthly, the modeling framework should maintain the well-formedness of the enterprise model and the consistency between different views opened by different stakeholders of the team (e.g. `BookCo` appears in multiple views of which one shows the value network level and another shows the company level). Table 1 summarizes this analysis.

The work presented in this article addresses the aforementioned challenges. It was initially developed as part of the SEAM method.