

A Semi-formal Evaluation of Architecture Design Based on Architecture Principles

Diana Marosin¹, Sepideh Ghanavati²

¹ Radboud University Nijmegen, Nijmegen, the Netherlands

² Texas Tech University, Lubbock, TX, USA

marosin.diana@gmail.com, sepideh.ghanavati@ttu.edu

Abstract. Enterprise architectures (EA) and principles have been defined multiple times in literature. The main focus is how architecture principles help fulfilling organization’s goals. There is a consensus on the guiding roles of principles and on the fact that architecture designs are limited by them. However, there are no (semi-)formal methodologies to truly evaluate the links between principles and design. There is also no method to evaluate miss-alignments between principles and design or to guide re-design in case the set of principles changes. In this paper, we make initial steps towards a semi-formal method for traceability and consistency checks between architecture principles and EA design. We aim to provide a mechanism that supports overall change management of the EA design when defining and refining the set of EA principles.

Keywords: *architecture design, architecture principles, GRL*

1 Introduction

In the current literature, there are multiple definitions for concepts such as “enterprise”, “enterprise architecture (EA)” and “enterprise architecture principles”. The different definitions of “EA principles” are presented in multiple research papers, and included in several literature studies such as [9, 5]. EA principles are similar to non-functional requirements (NFR) however, they are widely accepted as a term in practitioners’ dictionary. Due to the similarities between EA principles and NFR, we model principles with goal modeling languages.

In our research, we collaborate with multiple practitioners and use their case studies. One important observation is that architecture models are static and we cannot perform any formal analysis. To overcome this limitation, we introduced a semi-formal framework, called *Principles-based GRL* [1], that is based on Goal-oriented Requirements Language (GRL) [2]. By leveraging metadata for stereotyping and graphical representation, we developed a methodology to link principles to strategies and to evaluate their impacts [8]. We justified the relationships with goals and vision of the organization and we provided rationales for design decisions. With the help of what-if analysis, we analyzed what happens if EA principles support conflicting goals, if the set contains conflicting principles or if some principles support only partially the goals.

In this paper, we focus on the influences EA principles exhibit on the design of EA models. To that end, we aim to answer the following research question:

How to manage and evaluate the consistency between architecture principles and architecture design models?

In past, there were some efforts to map EA designs with EA principles through ArchiMate [6] and Motivation Model [4]. This mapping does not provide analysis of the consistencies, nor does it have a tool-support. In this work, we model ArchiMate's elements in Use Case Maps (UCM) which is part of the User Requirements Notation (URN) [7] and it is combined with GRL. Modeling architecture designs with UCM helps analyzing the consistency between the design and principles in a tool-supported manner (through URN links and scenario analysis functionality of jUCMNav [3]) which is not possible with ArchiMate.

We introduce our approach and background assumptions, in Section 2. Next, we present an extension of our framework with UCM elements, in Section 3. We apply the extension and present an evaluation of a model in Section 4. We conclude and present directions of future work in Section 5.

2 Background - Principle-based GRL Approach

Example - part 1: For a better understanding of the context of our work, we present an extract from the set of EA principles defined and used by “The Tax and Customs Administration” of a European country, further-on called *TaxAdministration*. In definition of a high-level EA principle, “We have an exclusive digital administration”, *TaxAdministration* defined four *Key Actions*, as follows: [1] The information in digital administrations are fast and trustworthy accessible where and when necessary for the work. [2] We make applications either specific for mass, automatized administrative processes or specific for activities that are focused on human interaction. [3] Where we administer digitally we also archive digitally. [4] We take measures so that the safety of our information is ensured.

Each *Key Action* is linked semi-formally to a different task or a (soft-)goal, such that it supports the realization of higher level goals of the organization, contributing to an *Added Value*. This situation is represented in Fig. 3. The mapping of EA principles to GRL elements was tackled in details in our framework [8]. The mapping presented in Table 1 contains ontological elements exist in the definition of EA principles. We defined stereotypes and OCL rules such that we preserve the essence of principles while formally representing them with GRL. This modeling effort results in the GRL model represented in Fig.3.

3 Framework Extension with UCM Scenarios

Example - part 2: In Section 2, we introduced an architecture principle of *TaxAdministration*. This principle is an extract from *Concern Architecture*, which contains a total of nine architecture principles and the corresponding strategy for new tasks related to digitalization of the organization. We also

Table 1: Mapping architecture principles constructs to GRL Intentional Elements (IE), adopted from [8].

EA principle el.	Stereotype value	GRL IE
Name	«Principle»	Softgoal (□)
Added value	«AddedValue»	Softgoal, Goal (◻)
Impact/restrictions	-	Qual. / quant. value of GRL links.
Key actions	«KeyAction»	Task (◻)
Preconditions	«Precondition»	Softgoal, Goal, Task, Resource (□)

investigated *Fundamental Architecture for Online Services*. This document is a specialization of *Concern Architecture* and provides guidelines and patterns on how to design and implement provisioning of online services. The representation is textual or modeled by ArchiMate.

In *TaxAdministration*'s architecture documents, the architecture instructions, such as *Provide Personal Information*, describe which functions are recognized and which data are needed for implementation.

After modeling and formalizing EA principles with GRL, we focus on analyzing how the impact of EA principles can be propagated to the design layer. We select one of the major architecture languages, ArchiMate - a graphical language, which offers an integrated way to describe and visualize different architecture layers, their relations and dependencies. We exploit the traceability links between GRL and UCM to perform different types of analysis such as checking the consistency between EA principles and EA design models. GRL aims at capturing business or system goals, alternative decisions and rationales behind goals while UCM models functional and behavioral structures using scenarios. In the remaining of this paper, we show how to model ArchiMate models with UCM and present an informal analysis of the resulted models.

Guidelines for Translating ArchiMate Models to UCM

We consider a scenario complete when all elements from ArchiMate were modeled with UCM elements. Therefore, we define the following rules: **[R1]** Choose an element from ArchiMate model and add it to the UCM scenario as a *start*-point. **[R2]** Concurrent behaviors retrieved from ArchiMate models are represented using UCM AND-fork. **[R3]** The mapping is complete when all elements and relations from ArchiMate were handled. Mark in UCM with *end*-point. **[R4]** Unite all paths resulted in the UCM scenario with an AND-joint.

In ArchiMate, the relation arrows go both in and out an ArchiMate node, however we create a directed UCM path between the *start* and the *end* point. To that end, we advise to have only arrows that exit from parents-nodes. To preserve the semantics of relations such as *realize* or *used by* from ArchiMate, we rephrase these relations in UCM. An example is represented in Fig. 1. We

define the following mapping rule and a set of stereotypes, corresponding to each relation type: **[R5]** Create directed paths in UCM.

Table 2: Translating ArchiMate Elements to UCM Elements and Stereotypes






ArchiMate Element	UCM Element / Stereotype
Function  Service 	Responsibility $\cdots \rightarrow \times \leftarrow \cdots$ $\ll\text{Service}\gg$ $\ll\text{Function}\gg$
Object 	Object  $\ll\text{Object}\gg$
Used by \leftarrow	Direction arrow $\cdots \rightarrow \cdots$
Access $\cdots \dashrightarrow$	$\ll\text{uses}\gg$ $\ll\text{access}\gg$
Architecture layer	Component 



Fig. 1: Comparison between ArchiMate’s *used by* and UCM’s $\ll\text{uses}\gg$ relations

In UCM, a responsibility is defined as a scenario activity representing something to be performed: operation, action, task, function, etc. [7]. Therefore, ArchiMate behavioral elements are modeled with UCM following rules 6-8: **[R6]** Functions and services from ArchiMate are modeled as responsibilities in UCM. **[R7]** Objects in ArchiMate are translated to objects in UCM. **[R8]** Each architecture layer is represented by a UCM component.

4 (Re)Evaluation of Design by Combining Principle-based GRL with UCM

Considering the EA principle represented in GRL in Fig.3 and the UCM scenario represented in Fig.2, we manually identify and create URN links between the two models. *Note that*, in a real cases, there are multiple EA principles at hand and a considerable higher number of EA designs. In our example, using just one instance of each concept, we can (at best) analyze the impact partially. Furthermore, the URN links and the evaluation are done a-posteriori. In reality, this tasks would be performed a-priori by enterprise architects, alongside the management board or involved stakeholders.

- We traced “We archive digitally” $\ll\text{KeyAction}\gg$ from the principle’s goal model (see Fig.3) to service “Archive message” $\ll\text{Service}\gg$ on the information layer of the architecture (see Fig. 2). The execution of this service increases the

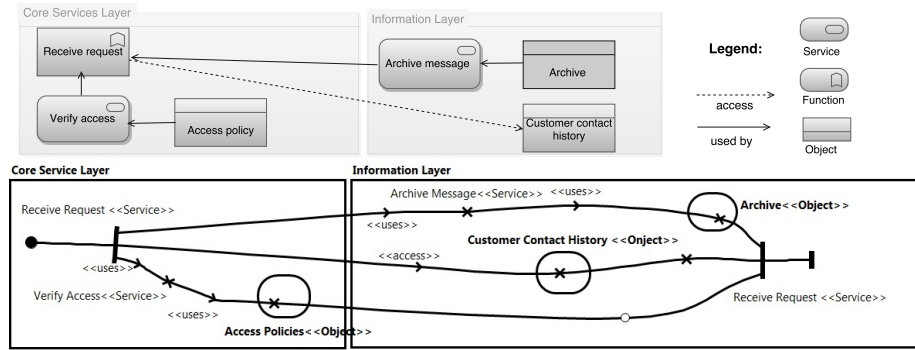


Fig. 2: Mapping of a Partial Architecture Model to a UCM Scenario

satisfaction level of the “We archive digitally” \llcorner KeyAction \gg intentional element, from an *unknown* value to *satisfied* (see Fig.3).

- We identified a link between “Measures to ensure safety of our information” \llcorner KeyAction \gg (see Fig.3) to “Verify access” \llcorner Service \gg (see Fig. 2). We consider this action beneficial, but not enough. In general more actions can contribute to the realization of a goal. In our example, we notice, we can improve the GRL evaluation from an *unknown* value to a *weakly satisfied*. This represents a positive increase of 25% from the initial evaluation.

URN provides guidelines for mapping GRL and UCM . We leverage these guidelines to provide the mapping between the Principle-based GRL and the extended UCM. However in future, we plan to perform an extensive analysis to better evaluate this mapping and provide better guidelines for traceability.

5 Conclusions

In the case of *TaxAdministration*, organization does not perform formal checks to assess if the EA models are in line with their EA principles. There are documents of architecture, connections between principles and goals, or refinements of principles, however, they are based on experience and observations.

In this paper, we proposed a method to link design solutions to EA principles. ArchiMate models cannot be executed. Consequently, it is impossible to automatize the check of actual behavior vs. the expected behavior foreseen by applying principles. By means of URN links, we are enabled to evaluate the efficiency of using and introducing a specific EA principle and to measure the (partial) compliance between EA principles and the implementation of architecture. Our approach provides a-posteriori evaluation, based on our interpretation, therefore, the real impact is still not fully perceivable. For future work, we aim to analyze the difficulty to translate the EA models using *Principle-based GRL framework*, and reason about the impact in the design phase. The complexity of models has to be evaluated keeping in mind the added value of the new models.

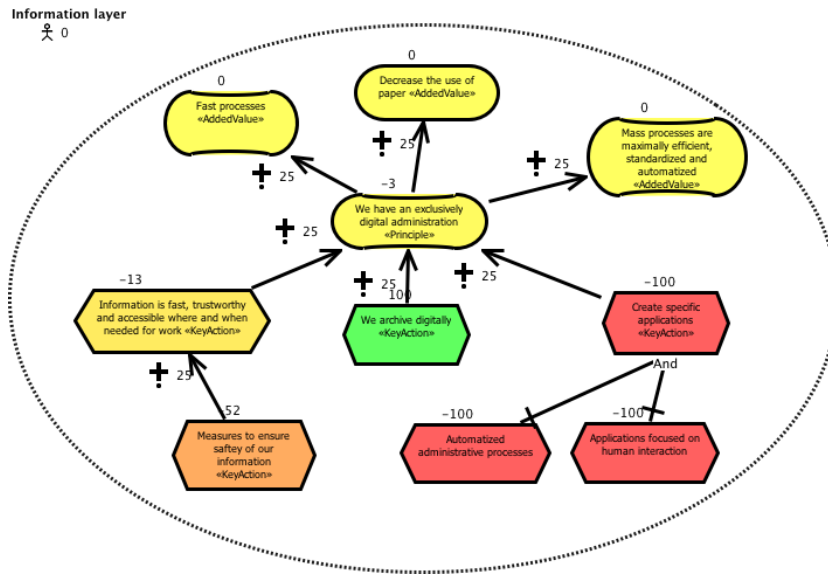


Fig. 3: Evaluation of EA Principle based on Execution of *Receive Request*.

Acknowledgements: We thank Saco Bekius and Michiel Borges for their collaboration and insights in the case study and our colleagues Marc van Zee and Erik Proper, for fruitful discussions and insights prior to this publication.

References

1. Principle-based GRL Fw. - <https://github.com/RationalArchitecture/eGovernment>.
2. D. Amyot, S. Ghanavati, J. Horkoff, G. Mussbacher, L. Peyton, and E. Yu. Evaluating goal models within the goal-oriented requirement language. *Int. J. Intell. Syst.*, 25(8):841–877, Aug. 2010.
3. D. Amyot et al. Towards advanced goal model analysis with jucmnav. In *ER Workshops*, volume 7518 of *LNCS*, pages 201–210. Springer, 2012.
4. F. J. Armour, S. H. Kaisler, and S. Y. Liu. A big-picture look at enterprise architectures. *IT Professional*, 1(1):35–42, 1999.
5. M. K. Haki and C. Legner. New Avenues for Theoretical Contributions in Enterprise Architecture Principles - A Literature Review. In *TEAR/PRET*, volume 131 of *Lecture Notes in Business Information Processing*, pages 182–197, 2012.
6. M.-E. Iacob, H. Jonkers, M. M. Lankhorst, H. A. Proper, and D. A. C. Quartel. *ArchiMate 2.0 Specification*. The Open Group, 2012.
7. ITU-T. Recommendation Z.151 (11/08): User Requirements Notation (URN) – Language Definition. <http://www.itu.int/rec/T-REC-Z.151/en>, 2008.
8. D. Marosin, M. van Zee, and S. Ghanavati. Formalizing and Modeling Enterprise Architecture (EA) Principles with Goal-Oriented Requirements Language (GRL). In *Proceedings of CAiSE 2016, Ljubljana, Slovenia, June 13-17*.
9. D. Stelzer. Enterprise Architecture Principles: Literature Review and Research Directions. In *ICSOC/ServiceWave Workshops*, LNCS, pages 12–21, 2009.