Realizing Traceability from the Business Model to Enterprise Architecture

W. Engelsman[1], R.J. Wieringa[2], M. van Sinderen[3], J. Gordijn[4], and T. Haaker[5]

[1] Saxion University of Applied Sciences,w.engelsman@saxion.nl
[2] The Value Engineers, roel@thevalueengineers.nl
[3] University of Twente, m.j.vansinderen@utwente.nl
[4] The Value Engineers & Vrije Universiteit Amsterdam, jaap@thevalueengineers.nl
[5] Saxion University of Applied Sciences, t.i.haakers@saxion.nl

Abstract. An enterprise architecture (EA) is a high-level representation of the enterprise, used for managing the relation between business and IT. In order to improve reasoning about the contribution of IT to the business, all elements of an EA should be traceable to the business model and vice versa. However, in practice this is not the case. Realizing this traceability would be useful because it would allow practitioners to reason about the contribution of IT to the Business Model of the organization. In addition to reasoning about cost structures and goal contributions of IT to the business, as is customary in EA, practitioners would also be able to reason about the contribution of IT to the value offerings of a business.
In this exploratory paper we investigate traceability between the EA, Business Model and Business Goals of an enterprise. We use ArchiMate as EA language and e^3-*value* as business modeling language, provide and motivate a hypothesis about how to realize traceability, and illustrate this with a real-world example. Our paper ends with a traceability hypothesis that will be further tested in future case studies.

1 Introduction

An enterprise architecture (EA) is a high-level representation of the enterprise, including business, IT and technology aspects. Large organizations maintain an EA in order to coordinate and steer IT projects and manage IT costs.

In our previous research we have clarified the relation between the business goals of the organization and its EA [5–7]. However, traceability to business goals captures only part of the motivation for an EA. High-level strategic goals are elaborated in a business model and the business model in turn motivates design choices in the EA [2]. In other words, an EA should not only be used to manage IT costs but also to manage the contribution of IT to the value offerings of an enterprise. In this paper we extend our work on traceability with a hypothesis about traceability to business models. One could say that in this traceability relationship, the business model provides us with puzzle pieces and the EA will put these pieces together [17].

As enterprise architecture modeling language we choose ArchiMate [15], both because it is well-known and because we have been involved in its development [5].

We take an ecosystem approach to business models, which means that we see a company as part of a network of companies that coordinate to produce a value proposition. We use e^3-*value* as ecosystem business modeling language [9]. e^3-*value* can be used to model the value exchanges in an ecosystem needed to jointly produce a value offering to a customer, and to analyze the long-term commercial viability of this. The current paper focusses on the implications of an ecosystem business model for the internal EA of an enterprise.

In this paper we show how the EA of one of the enterprises in an ecosystem, specified in ArchiMate, can be aligned with an ecosystem business model, specified in e^3-*value* . We expect that realizing this traceability provides us with the following advantages:

- Tracing not just to the business goals, but actually to quantifiable elements from the business model allows for better reasoning, especially in value networks where goals of different stakeholders need to be brought together to create a shared perception of the ecosystem, and to analyze the commercial viability of the ecosystem.
- To assess which projects that implement the architecture have the most business value in terms of contribution to the enterprise business goals and the ecosystem business model.

In section 2 we analyze related work and, based on this, introduce a hypothesis about traceability links between ArchiMate and e^3-*value* models. In section 3 we test this initial hypothesis on a real-world example and, based on this application, refine our initial traceability hypothesis in the form of a metamodel. In section 4 we discuss the validity of our research method and outline some future research.

2 Value Modelling, Goal Modelling and Enterprise Architecture Modelling

There is some existing work done trying to link business models, goal modelling and EA. Gordijn et al. [11] propose a method to combine i* with e^3-*value* , with no focus realizing on traceabililty. Andersson et al. [3] describe the alignment of business models and goals. They have developed templates that align goal statements with value propositions. Meertens et al. [14] propose similar work, but instead of using e^3-*value* they provide a mapping from the Business Model Canvas to ArchiMate. Pessoa et al. [16] developed a method for requirements elicitation for business models using an early version of the motivation extension of ArchiMate. Gordijn et al. [10] propose a method for requirements engineering for e-services. Aleda et al. [1] propose adaptations of ArchiMate to incorporate value modelling, but does not try to create traceability between different languages and the concepts introduced are less detailed than those using e^3-*value*

. In general, the major difference of our work with related work is our focus on traceability through meta-models.

Our earlier proposal to extend ArchiMate with a goal modelling language called ARMOR [5–7] is now part of Archimate 3.0 [15]. In follow-up empirical studies we observed that most of the concepts of ARMOR were conceptually too complex for practical use, and we proposed a simpler version of the language that is usable in practice, called ARMOR-light [7]. In ARMOR-light we only use the notions of stakeholder and goals. When a goal is realized by an element from ArchiMate then it is considered a requirement, similar to KAOS [4]. In this paper we will use ArchiMate 3.0 as if it only contains the constructs of ARMOR-light.

ArchiMate models have a business layer, an application layer, and a technology layer, that have traceability links among them. To realize traceability to e^3-*value* models, we need to link the business layer of an ArchiMate model to e^3-*value* models. Within the business layer, the Business Service is used to expose behavior and value of the organization to the environment. This is where we expect to find the link between ArchiMate and e^3-*value* .

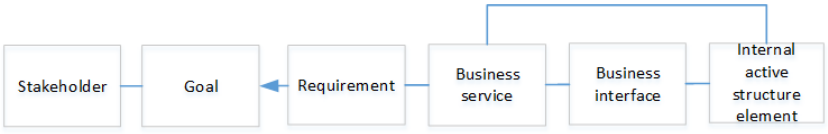

Fig. 1. Metamodel of ARMOR-Light with part of the metamodel of ArchiMate [7]. The lines represent many to many relations, the arrow represents a subset.

Figure 1 shows part of a meta-model of ARMOR-light and ArchiMate. For clarity reasons we have omitted the application and technology layers and a large part of the business layer. Requirements are the subset of goals allocated to a business service. Goals not allocated to an EA element are ends that a stakeholder wishes to achieve. In ArchiMate an internal active structure element is an abstraction of any actor or specialization thereof; e.g. roles, actors, collaborations, etc. A business service is the externally visible behavior of an internal active structure element. It exposes its behavior over a business interface of the internal active structure element, e.g. the sales channel.

Ik heb het plaatje in overeenstemming met fig. 6 gemaakt: Een iets ander deel van https://pubs.opengroup.org/architdoc/chap08.html. En de tekst aangepast. –Roel

Onderstaand stukje snnap ik niet. Wat is een internal active structure? Hoezo zijn sommige goals ends? Zijn die andere van means? –Jaap

In Figure 2 an educational e^3-*value* model is presented, annotated with the name of the modeling constructs, which we discuss below. In e^3-*value* , an actor is some entity capable of performing value activities, e.g. a business, department or partner. In the example, the book store is an actor. A special case of an actor is the market segment (e.g. the reader or the publisher). A market segment models that are many actors of the same kind. In e^3-*value* this means that all actors in a market segment assign economic value precisely in the same way. A value activity (not shown in the example) is a task performed by an actor which can lead to a positive net cash flow. The value activity differs from activities in process models in e.g. the BPMN. Value activities should be profitable while in

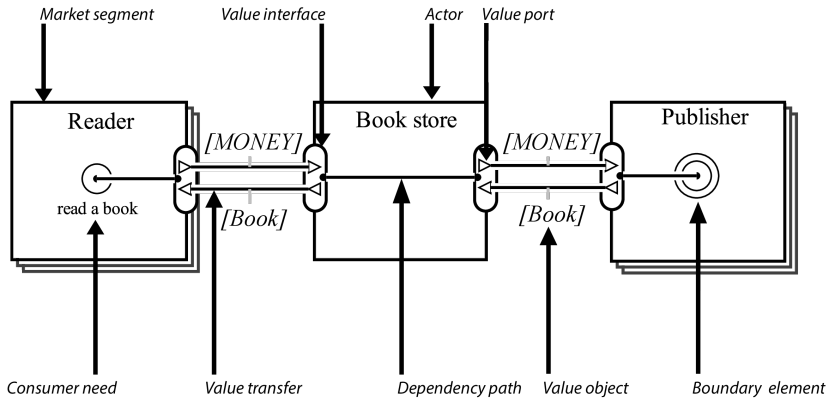

Fig. 2. Educational e^3-*value* model

BPMN it is perfectly allowed to include activities that only cost money. A value interface represents what the actors offers and requests to/from its environment in terms of value objects. Value objects are things that are perceived by at least one actor as of economic value. A value interface consists of at least one ingoing and one outgoing port, through which the actor requests or offers value objects from or to its environment. The value interface models (1) the notion of economic reciprocity and (2) bundling. Economic reciprocity is the idea that someone only offers something of value, of something else of higher economic value is obtained in return. In the example, the book is exchanged for money, hence the transfers are economically reciprocal. Bundling is the case where it is only possible to offer or obtain value objects in combination. Value ports between actors are connected by means of value transfers, which represent the willingness of actors to exchange things. Internally in an actor, there is the dependency path, which shows how value objects exchanged via a value interface require or assume exchanges via other value interfaces of that same actor. For example, the sale of a book by the book store requires that this store obtains the book from a publisher. The boundary element of a dependency path indicates the boundary of our modeling interest. Any further transactions that take place in the real world to satisfy the consumer need are not included in our model.

Figure 3 shows the meta-model for actors (left) and dependency paths (right) in e^3-*value* [8].

Based on these two metamodels, we formulate our initial hypothesis about traceability between ArchiMate and e^3-*value* concepts in Table 2.

3 Application: Cirque du Soleil

Volgorde in fig 3 moet zijn: Visitor, Digital customer, CdS, Ticket Office, Samsung. Dan correspondeert het beter met Fig. 4. "Distribute live performance" -> Digital distribution of VR show"en "Distribute tickets"-> "Sell tickets –Roel

3.1 The example

Cirque du Soleil, based in Montreal, is known internationally for its innovative form of circus production. Cirque was one of the first to reinvent the circus

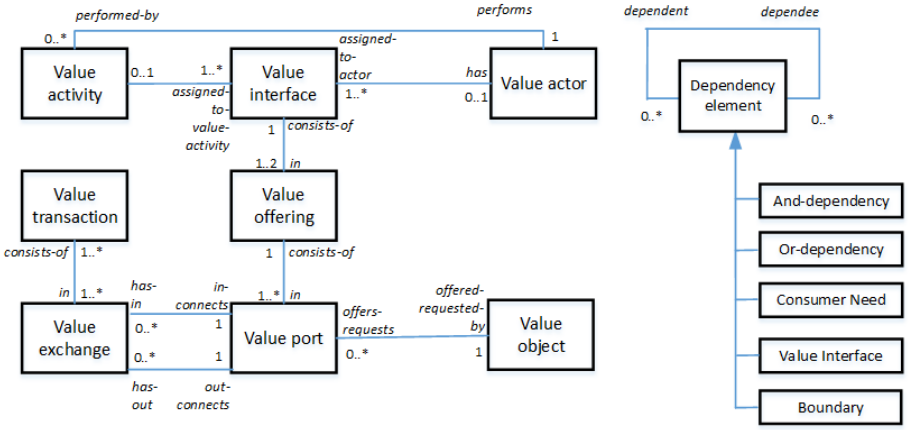

Fig. 3. Metamodel of actors and dependency paths in e^3-*value* [9]

ArchiMate	e^3-*value*	Argument
Stakeholder	Actor	An e^3-*value* actor is always an ArchiMate stakeholder, by definition. The other way around is not guaranteed.
Goal	Customer need	Customer needs are customer goals. Goal models identify and refine customer needs.
Business Actor	Actor	Both are essentially the same thing. An actor in e^3-*value* performs activities that produce value. in ArchiMate actors perform behaviour as well.
Business Service	Value activity	Both concepts denote externally visible behavior performed by an actor and made available through an interface.
Business interface	Value interface	Both are the interfaces in which the behavior is accessed.

Table 1. Our initial hypothesis about correspondences between concepts in ArchiMate and in e^3-*value* .

production, without animal usage, but with a focus on artistic human performances [13]. We collected information about this example from public sources on the internet [12, 19] supplemented with some assumptions to round out the example. We will show how Cirque du Soleil offers live shows and virtual shows, offered in an attractive location and through a VR device, respectively. Tickets for the live shows are sold by an independent ticket office. The VR shows are distributed by Samsung.

3.2 Goal model, business model and EA model

stukje weggelaten om ruimte te besparen.–Roel

We start by constructing a goal model in ArchiMate (figure 4). For illustration purposes we have restricted ourselves to one goal per relevant actor. A goal model like this is often constructed in the strategic phase of EA development, similar to TOGAF's preliminary and vision phases [18]

Figure 5 contains the e^3-*value* model that illustrates the value adding activities. Value activities are represented by rounded rectangles inside an actor. In our example Samsung enters into a collaboration with Cirque Du Soleil to distribute the VR media of the circus performance to customers. An external ticket office is used to offer a ticketing service. For example, Cirque du Soleil wants to perform a show and Samsung wants to distribute performances.

Customers are represented by two separate actors, Visitor and Digital Customer. Visitors have a need to enjoy a live artistic show, and satisfy this need by paying Cirque du Soleil for performing their value activity. Cirque du Soleil hires a ticket office to sell tickets. The inter-actor transactions and the intra-actor dashed lines form a dependency path in e^3-*value* models, connecting a consumer need with all transactions in the ecosystem needed to satisfy the need. The customer need to enjoy a show from home is satisfied by a similar dependency path.

Figure 6 shows an ArchiMate model of the EA for Cirque du Soleil. We have identified two different main Business Services: the Circus Performance Service and the Digital Distribution Service. These two business services correspond to the value activities of the e^3-*value* model from the actors Samsung and Cirque du Soleil. The ArchiMate model also contains four business actors, where Samsung and Cirque du Soleil collaborate together to deliver the digital distribution service. To model the different roles of the customers we have chosen to model the digital customer and the visitor as seperate roles.

The same can be seen with the ticket office, they collaborate (the business collaboration) to provide the ticketing service. Since ArchiMate allows for more detail in the modelling of the business services, design decisions like the composition of the ticketing service in the circus performance service are represented here. It is also possible that these translate to supporting internal business services like the recording service. Before you can distribute a show you do need some sort of recording service. This is not necessarily a value adding activity and therefore not visible in the e^3-*value* model.

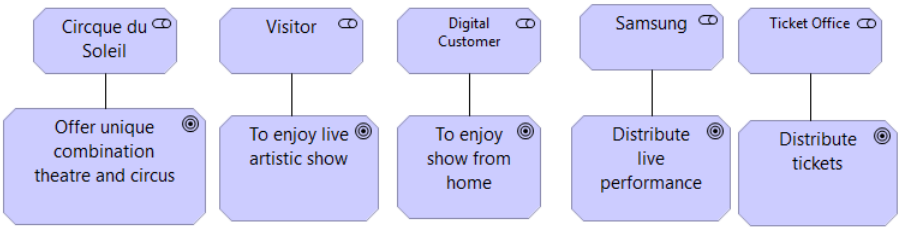

Fig. 4. Partial Cirque du Soleil Goal Model

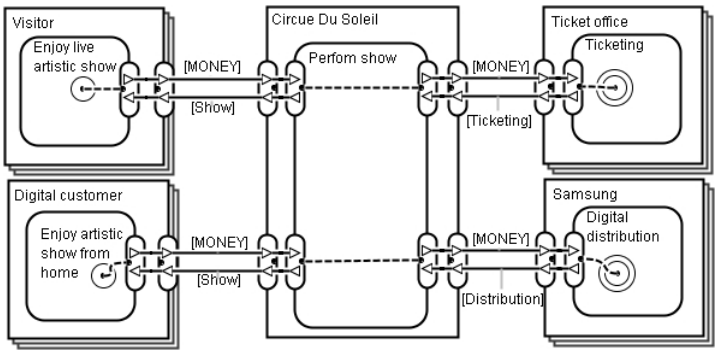

Fig. 5. *e^3-value* model of Cirque du Soleil

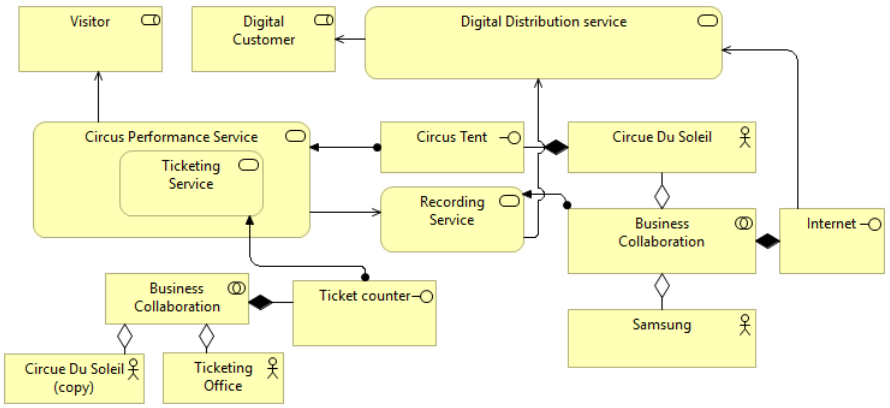

Fig. 6. ArchiMate model of the business layer of Cirque du Soleil

3.3 Observations

Linking ArchiMate goal models to e^3-value *models.* First, we see that in our example, goals in the goal model correspond to value activities of business actors in *e^3-value* . We believe this to be a general rule for strategic goals. There are goals at every level of the organization, but only strategic goals will be relevant for a business model and may appear there as value activities.

In addition, consumer needs in our example correspond to consumer goals in the stakeholder model. This leads us to the following three refinements of our initial two hypotheses about the correspondence between ArchiMate goals models and *e^3-value* business models:

- Stakeholders with strategic goals correspond to actors in an *e^3-value* model.
- Value activities in an *e^3-value* model correspond to lower level goals in a strategic goal model.
- Consumer needs in an *e^3-value* model correspond to lower-level consumer goals in a strategic goal model.

Linking e^3-value *models to ArchiMate EA models.* In our example all the value activities that do not have a consumer need attached to them correspond to business services in ArchiMate.

Actors in the *e^3-value* model correspond to business actors in the ArchiMate model. This may not be true in general as ArchiMate also contains the concept of a role. An *e^3-value* actor may correspond to a role in the ArchiMate EA rather than to a business actor. Future research should provide clarity about this.

The value interfaces in *e^3-value* map onto the business interface in ArchiMate. For example, the four value interfaces from Samsung to Cirque du Soleil translate to a single business interface in ArchiMate.

Finally, an *e^3-value* dependency path connects transactions among different actors. This may be mapped to business collaborations in an ArchiMate model. Whether this is true in general must be shown by future case studies.

This leads us to the following refinements of our initial hypotheses about the correspondence between ArchiMate EA models and *e^3-value* business models:

- *e^3-value* actors map to business actors and possibly roles in an ArchiMate EA models.
- *e^3-value* activities map to ArchiMate business services.
- *e^3-value* value interfaces map to ArchiMate business interfaces
- An *e^3-value* dependency path may map to a business collaboration in ArchiMate.

tabel weggelaten, staat al in de bulleted lists hierboven – Roel

Figure 7 summarizes our traceability rules. This meta-model is divided into three different layers:

- The strategic layer where we find stakeholders and goals,
- he value layer, where we see the value adding acitvities and
- the technical layer where we find the designs of the organization in an EA.

Integrating *e*^{3}*-value* into ArchiMate therefore happens between the Motivation layer and the Business Layer. This results in traceability from stakeholder to actor and ArchiMate equivalents. Conversely, from a business service we can trace to value activities and then to the value offerings, and directly and indirectly to goals.

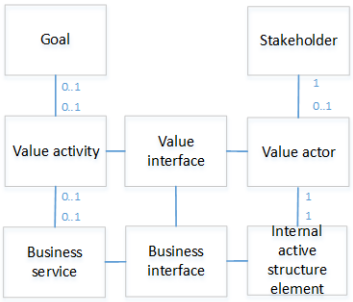

Fig. 7. Combined traceability model. All relationships are many-many unless otherwise stated.

4 Discussion and Future Work

This work is in its early stages and our current hypothesis is based on an analysis of metamodels plus an application to a single example. We cannot claim generalizability based on this.

To test generalizability, we will restrict our research to *e*^{3}*-value* and Archimate. Generalizability to other languages therefore remains an open issue. However, we will investigate generalizability to other cases analyzed in *e*^{3}*-value* and Archimate as a next step.

Within this scope, we need to refine the hypothesis by doing more complicated real-world case studies. What is the exact meaning of the relations in our proposed integrated metamodel? What is the relation between value activity and goal? Could it be a realization or specialization?

We will test usability of our hypothesis in experiments like we did with ARMOR [5–7]. Utility in practice will be investigated by means of opinion research, e.g. a focus group of practitioners. A final step is to create a tool-supported method for designing an EA based on a business model, and for extracting a business model from a given EA.

References

1. A. Aldea, M.E. Iacob, J. van Hillegersberg, D. Quartel, and H. Franken. Modelling value with archimate. In *International Conference on Advanced Information Systems Engineering*, pages 375–388. Springer, 2015.

2. R. Alt and H.-D. Zimmermann. Preface: introduction to special section–business models. *Electronic markets*, 11(1):3–9, 2001.
3. B. Andersson, M. Bergholtz, A. Edirisuriya, T. Ilayperuma, P. Jayaweera, P. Johannesson, and J. Zdravkovic. Enterprise sustainability through the alignment of goal models and business models. In *Proceedings of 3rd International Workshop on Business/IT-Alignment and Interoperability (BUSITAL'08) CEUR Workshop Proceedings*, 2008.
4. A. Dardenne, A. van Lamsweerde, and S. Fickas. Goal-directed requirements acquisition. *Science of Computer Programming*, 20(1–2):3–50, 1993.
5. W. Engelsman, D. A. C. Quartel, H. Jonkers, and M. J. van Sinderen. Extending enterprise architecture modelling with business goals and requirements. *Enterprise information systems*, 5(1):9–36, February 2011.
6. W. Engelsman and R. Wieringa. Goal-oriented requirements engineering and enterprise architecture: two case studies and some lessons learned. In *Requirements Engineering: Foundation for Software Quality*, pages 306–320. Springer, 2012.
7. W. Engelsman and R. J. Wieringa. Understandability of goal-oriented requirements engineering concepts for enterprise architects. In *Advanced Information Systems Engineering (CAiSE), 26th international conference*. Springer, 2014.
8. J. Gordijn. E-business value modelling using the e3-value ontology. In *Value creation from e-business models*, pages 98–127. Elsevier, 2004.
9. J. Gordijn and J.M. Akkermans. *Value Webs: Understanding E-business Innovation*. The Value Engineer, 2018. http://www.thevalueengineers.nl.
10. J. Gordijn, P. Van Eck, and R. Wieringa. Requirements engineering techniques for e-services. *Service-Oriented Computing: Cooperative Information Systems Series*, pages 331–352, 2009.
11. J. Gordijn, E. Yu, and B. Van Der Raadt. E-service design using i* and e^3value modeling. *IEEE software*, 23(3):26–33, 2006.
12. W. Chan Kim and R.A. Mauborgne. *Blue ocean strategy, expanded edition: How to create uncontested market space and make the competition irrelevant*. Harvard business review Press, 2014.
13. D. Leslie and N.M. Rantisi. Creativity and place in the evolution of a cultural industry: the case of Cirque du Soleil. *Urban Studies*, 48(9):1771–1787, 2011.
14. L.O. Meertens, M.-E. Iacob, L.J.M. Nieuwenhuis, M.J. van Sinderen, H. Jonkers, and D. Quartel. Mapping the business model canvas to ArchiMate. In *Proceedings of the 27th annual ACM symposium on applied computing*, pages 1694–1701. ACM, 2012.
15. The Open Group. *ArchiMate 3.0.1 Specification*. `http://pubs.opengroup.org/architecture/archimate3-doc/`.
16. R.M. Pessoa, M. van Sinderen, and D.A.C. Quartel. Towards requirements elicitation in service-oriented business networks using value and goal modelling. In *ICSOFT (2)*, pages 392–399, 2009.
17. D.J. Teece. Business models, business strategy and innovation. *Long range planning*, 43(2-3):172–194, 2010.
18. The Open Group. *TOGAF Version 9*. Van Haren Publishing, 2009.
19. Computer Weekly. Case studies; the sytems behind the shows. `https://www.computerweekly.com/feature/Case-studies-The-systems-behind-the-shows`, 2006.