Jaap Gordijn,
Maya Daneva (Eds.)

BUSITAL'07
Second International Workshop on Business/IT Alignment and Interoperability

Workshop at

CAiSE'07

The 19th International Conference on
Advanced Information Systems Engineering
Trondheim, 11-15 June, 2007

Organization

Organizers:
J. Gordijn, Free University of Amsterdam, The Netherlands
M. Daneva, University of Twente, The Netherlands

Program Committee:
E. Dubois, Public Research Centre Henri Tudor, Luxembourg
C. Ebert, Vector Consulting, Stuttgart, Germany
P. Johannesson, KTH Stockholm, Sweden
J. Krogstie, NTNU, Trondheim, Norway
K. Lyytinen, Case Western Reserve University, USA
M. Petit, University of Namur, Belgium
Y. Pigneur, University of Lausanne, Switzerland
H. Weigand, Tilburg University, The Netherland
A. Wegmann, EPFL Lausanne, Switzerland
R.J. Wieringa, University of Twente, Enschede, The Netherlands
C.Woo, University of British Columbia, Vancouver, Canada
E. Yu, University of Toronto, Canada

Preface

Welcome to BUSITAL 2007 – the workshop on business/IT alignment and interoperability!

The goal of the BUSITAL workshop series is to provide a forum in which leading researchers and practitioners from around the world can discuss and advance the state-of-the-art research and practice in business/IT alignment. The specific goal of BUSITAL'07 is to investigate how well established and emerging conceptual modeling methods, techniques, and tools fit in solutions to confront the challenge of maintaining mutual alignment between business needs and IT assets.

Within the scope of CAiSE, which is the development, maintenance, procurement, and use of information systems, business and IT alignment is a critical "early stage" exercise to understand how information systems contribute to business strategy and to set directions for the downstream development and maintenance processes. BUSITAL'07 participants and position papers are thus expected to show how ideas based in value-driven thinking, broadly construed, can be adapted to improve the outcomes of system development initiatives. We focus on both: (i) the needs for alignment among the business strategy (business goals and business models), enterprise modeling (business processes and organization infrastructure), and information systems (infrastructure and applications), and (ii) the identification and assessment of suitable conceptual modeling methods and techniques that fit as the glue for making this alignment effective. At the workshop, evidence is provided in the form of short position papers. Each one deals with (i) a specific business/it alignment problem, (ii) a value-oriented approach to confronting it, and (iii) analysis, data or other evidence to support the proposed solution approach. The BUSITAL workshop discusses ongoing research, with a reasonable degree of plausible theoretical or practical utility.

BUSITAL'07 would not have been possible without the efforts and expertise of a number of people who volunteered their time and energy to help make this workshop a success. We would like to thank all our Program Committee members, the BUSITAL'07 speakers, and the CAiSE organizers for contributing to this success.

J.Gordijn, M. Daneva

Table of Contents

Using Strategic Goal Analysis for Enhancing Value-based Business Models

Birger Andersson, Maria Bergholtz, Ananda Edirisuriya, Tharaka Ilayperuma, Paul Johannesson, Jelena Zdravkovic

Department of Computer and Systems Sciences
Stockholm University and Royal Institute of Technology
Forum 100, SE-164 40 Kista, Sweden
{ba, maria, si-ana, si-tsi, pajo, jzc}@dsv.su.se

Abstract. Lately business models have been recognized as a foundation for design of operational business processes. The motivation of a business model can be found in the goals of an enterprise which are made explicit in a goal model. This paper discusses the alignment of business models with goal models and proposes a method for constructing business models based on goal models. The method is based on a template and rules based approach. The outputs are business models that conform to the explicit goals of an enterprise. Main benefits are uniform goal formulations, well founded business model designs, and increased traceability between the models. A case study from the health sector is used to argument the way we ground and apply our proposed method.

1 Introduction

Business modeling can be used as a starting point for an enterprise when setting out to model its processes. A natural way of working for a business analyst is to first establish in a business model what kinds of business elements, like actors, resources and resource exchanges that exist and later determine how they are to interact with each other in activities and processes. A number of ontologies [1], [2], [3] have been developed in order to precisely state what to include in a business model.

A goal model captures the purpose of a business. The motivation behind the design of a business model can be found in the goals of the business. For example, if a goal is formulated as “we shall outsource our delivery service”, then a transport agent shall be included in the business model.

Goal and business models are parts of a chain of models, together with process models, that describe different aspects of a business [4]. A common view is that:

– Goal models are used in the earliest phases of business and information systems design, helping in clarifying interests, intentions, and strategies of different stakeholders answering to the "why" of the business.
– Business models give a high level view of the activities taking place in and between organizations by identifying agents, resources and the exchange of resources between the agents. So, the model focuses on the "what" of a business.

– Process models focus on the "how" of a business, dealing with operational aspects of business communication, including control and data flow, and message passing.

Health care is one form of business that in recent years become increasingly process-oriented. Therefore analyzing and describing care taking in terms of the goal-business-process chain is valuable much in the same way as other businesses. One purpose of such an analysis is to align the care taking unit's IT systems with the explicit goals of the involved actors (e.g., patient, hospital, or financial institutions). In this respect, first the gap between goal model and business models must be bridged. The relation between goal and business models is discussed in [6], but there with the purpose of getting an understanding of business strategies.

We propose a method that helps designing a business model based on the explicit goals of actors. It extends the work of [5] which is based on using templates and rules for transforming one business model into one that also takes goals into consideration. The method reported here furthermore allows for a separation of concerns by letting the goals/means definitions be based on patterns that distinguish between strategy and tactics (i.e. how the strategy will be carried out), allowing for the tactic details of a goal model to be tailored to the situation at hand by selecting appropriate patterns.

The rest of the paper is organized as follows. Section 2 introduces a running health care example and in parallel describes the chosen business modeling notation. Section 3 addresses how goal models can be related to business models by means of a number of templates for structuring information. In Section 4 a method that transforms a given business model into a new business model based on a goal model is suggested. Its main points are illustrated through an application on the running health care example. Section 5 concludes the paper and gives suggestions for further research.

2 Example Case in e^3*value* ontology

For the purpose of illustrating a basic business model, a small running case taken from an eye-healthcare domain is introduced. The business model formalism used in this paper is that of e^3*value*, an established business model ontology which is widely used for business modeling in e-Commerce [2]. The e^3*value* ontology aims at identifying value exchanges between business actors. The basic concepts in e^3*value* are *Actors*, *Value Objects*, *Value Ports*, *Value Interfaces*, *Value Activities* and *Value Exchanges*. An Actor is an economically independent entity. Examples: enterprises, end-consumers. A Value Object is something that is of economic value for at least one Actor. Examples: cars, Internet access, services such as health care treatment. A Value Port is used by an actor to provide or receive Value Objects to or from other Actors. A Value Port has a direction, in (e.g., receive goods) or out (e.g., make a payment) indicating whether a Value Object flows into or out of the actor. A Value Interface consists of in and out ports that belong to the same actor. Value Interfaces are used to model economic reciprocity. A Value Exchange is a pair of value ports of opposite directions belonging to different actors. A Value Activity, finally, is an operation that could be carried out in a profitable way for at least one actor.

Figure 1 depicts an e^3*value* model of the eye-healthcare case (the model is an excerpt of a larger case defined in the REMS project [7]) that will be used as a running example throughout the paper. Actors are shown by rectangles, value activities by rounded rectangles, value ports by triangles, value interfaces by oblong rectangles enclosing directed value ports, and value exchanges as lines between value ports with the names of value objects as labels.

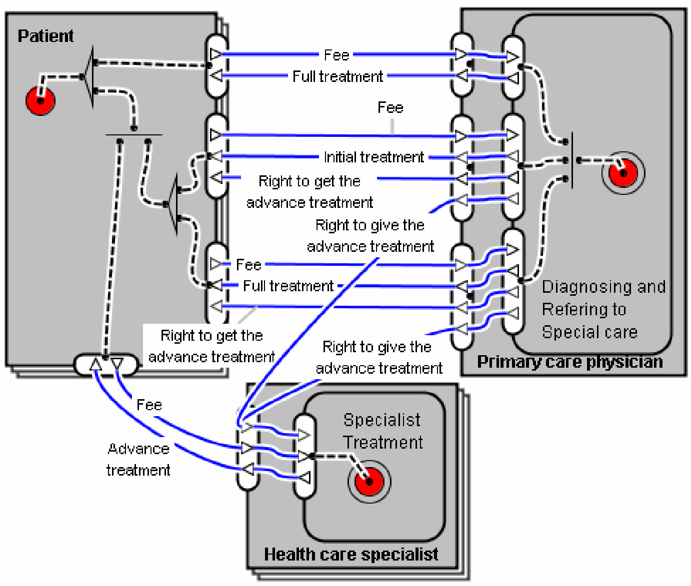

Fig. 1. *e^3 value* model for the eye-care case

The model depicts the value exchanges taking place between three actors: a patient, a primary health provider and a specialist at a clinic. Having a problem with her eye(s), the patient contacts the local primary care physician, in order to get a treatment. If necessary, the primary care physician refers the patient to an eye specialist clinic. The resources (i.e., value objects) that the patient gets are an initial or a full treatment, and the referral, if necessary. In return, the physician gets the patient fee. In relation to this, exchanges of values also occur between the physician and the specialist eye-clinic (e.g. the right to give treatment is transferred from primary care taker to specialist clinic) as well as between the patient and the specialist (the patient gets special treatment in return for a fee).

3 Goal Model and Means Templates

In this paper we consider the use of a goal model approach that supports analysis of strategic business goals such as i* [8], or the Business Motivation Model (BMM) [9]. The i* technique focuses on modeling strategic dependencies among business actors, goals, tasks and resources. In this study, the main focus is set on establishing a relationship between goal and value components. For this, we use the BMM, as the technique primarily focuses on the business states a principle actor wish to achieve, as well as on the actions that will enable the achievement of those states. The technique relies on the use of three major concepts – *Ends*, *Means,* and *Influencers*. An End is a goal the principal actor seeks to accomplish, without any indication of how it will be achieved. A typical goal of a car-rental company could be “to provide leading customer service”. A Means represents a course of action that is used to achieve Ends (goals). Thus, for the previously given goal example, a means for providing a leading customer service can be “hire experienced customer service personnel”. When a goal is described in a highly abstract manner, it is common to first divide it into sub-goals down to the level where they can be supported by concrete means. An Influencer is anything that may impact the achievement of means and goals. The impact of an influencer may be categorized in different ways - a simple and commonly accepted classification is as strength or weakness for internal influencers (for instance, resources or infrastructure), and as opportunity or threat for external ones (such as customers, competitors, environment, technology, etc.) [10].

A common problem in goal modelling is that goals are difficult to formulate, that is, the formulations of goals and means often become loose and highly abstract. This is because goals typically range from the value propositions of an enterprise to general goals of economic sustainability. We suggest overcoming this problem by expressing goal model elements in terms of business model notions. Business models describe the use and exchanges of resources that are of economic value for the participating actors. It means that the goals are to provide actors with desired resources. A resource may have properties and associations to other objects, such as the number of shops accepting a credit card, which are modelled by means of features [11]. After surveying a number of goal models, we found that the means in these models concern the acquisition, production, use, or provisioning of resources, which may be described using business model notions. These observations motivate the following rules for formulating goals, means and influencers in BMM:

A Goal is expressed as *a desired condition on one or more features of a resource, from one particular actor’s point of view*. One example from the eye-care case is “The diagnosis (resource) shall be correct (feature)” (see Figure 2).

A Means is expressed as *a course of action on one or more business model components* (value object, value activity, or actor) realising the desired conditions on resources stated by one or more goals. Means play a key role in aligning a business model with a goal model. As we stated earlier, means addresses the archetype business activities and therefore it becomes possible to formulate them according to a small number of templates. The following structure is used for the templates:

– Each template has two parts, one compulsory and one optional.
– With the compulsory part, the goal modeler describes the main course of action that is actually a *strategy* for how to realize one or more goals.
– The optional part describes an appropriate *tactics* that could be carried out in order to fulfill the compulsory part, that is, the strategy.

Formally, a means is defined as:

Strategy (course of action) [BY tactics_1 (course of action) OR BY tactics_2…], where the course of action is a triplet, <Action, Value Object, Actor/Value Activity>

For instance, a course of action may state “offer value object to (actor | market segment)”. When describing a means, the goal modeler is not obliged to articulate a tactics, he will only do so if the elicited goals lead to use of particular tactics. The following six means templates have been identified:

1. Offer value $object_1$ to $actor_1$ [as compensation for value $object_2$ from $actor_2$] [BY procure value $object_1$ from $actor_3$ [as compensation for value $object_3$ to $actor_3$] OR BY produce value $object_1$]
2. Stop offer value object to $actor_1$ [BY stop procure value object from $actor_2$ OR BY stop produce value object]
3. Use value object in value activity [BY procure value object from actor/market segment OR BY produce value object in value activity]
4. Stop use value object in value activity [BY stop procure value object from actor OR BY stop produce value object]
5. Outsource [fraction of] production of value object to actor
6. Insource [fraction of] production of value object from actor

Note that in the given list, compensation is used to describe a returned value object, if default is not used (i.e. money, fee).

An Influencer is expressed as a condition that leads to support, refinement or removal of one or more means or goals (see Figure 2 for an example).

Using the outlined definitions for goals, means and influencers, Figure 2 shows a partial goal model for the eye-care business scenario from Section 2. The analysis regards the patient's demand to get a right treatment, which is therefore articulated as a top-goal for the primary-care physician and the specialist.

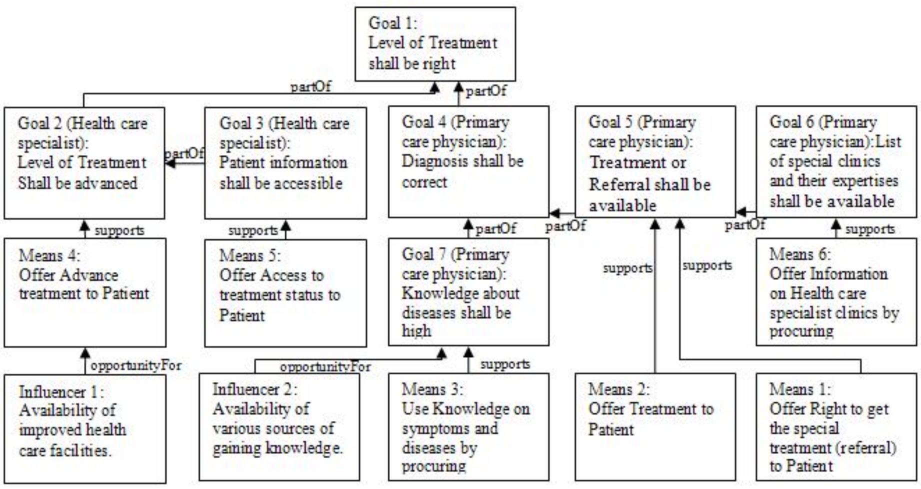

Fig. 2. A goal model for the eye-care case.

4 Creating a Goal-based Value Model

The means elicited in Figure 2 are described in the form of given templates, and as such, provide a basis for structuring certain value components in the business model.

In the following, we propose a method that takes as input a business model and a goal model and produces a new goal-enhanced business model conforming to the goal model. The main instruments used in the method are the means templates from the previous section and the patterns and action rules introduced below. Due to space restrictions, only the first template from Section 3 will be discussed in full detail.

Transformation rule patterns and associated actions

1. Offer value $object_1$ to actor1 [as compensation for value $object_2$ from $actor_2$] [BY procure value $object_1$ from $actor_3$ [as compensation for value $object_3$ to $actor_3$] OR produce value $object_1$].

This template addresses the business activity of exchanging value objects between actors. The compulsory part (strategy) deals with providing a value object to an actor. The first optional part allows a specification of what value object will be received as compensation for providing the first value object. The second optional part addresses the tactics and offers two alternatives: procure the value object from someone (possibly in turn for another value object) or else produce the value object yourself.

Below are the set of patterns associated with this means template. The value modeler will choose one of these patterns and furthermore apply the set of actions associated with that pattern.

a. Offer value $object_1$ to $actor_1$ and receive value $object_2$ from $actor_1$
b. Produce value $object_1$ and offer value $object_1$ to $actor_1$ and receive value $object_2$ from $actor_1$
c. Procure value object1 from $actor_2$ by providing value $object_3$ to $actor_2$ and offer value $object_1$ to $actor_1$ and receive value $object_2$ from $actor_1$
d. Stop Produce value $object_1$ and Procure value $object_1$ from $actor_2$ by providing value $object_3$ to $actor_2$ and offer value $object_1$ to $actor_1$ and receive value $object_2$ from $actor_1$
e. Stop procure value $object_1$ from $actor_2$ and Produce value $object_1$ and offer value $object_1$ to $actor_1$ and receive value $object_2$ $actor_1$

The actions describe how each pattern changes the current e^3-Value model. In the following we illustrate the actions that are associated with the pattern *a*.

If pattern *a* then

Add new actor if $actor_1$ doesn't exist.

Add new value exchange to offer value $object_1$ to $actor_1$ and connect this new value exchange to an existing value activity.

If necessary add new value exchange to receive value $object_2$ from $actor_1$

Method Application

Means 6 – Offer Information on Health care specialist clinics by procuring

- Step 1. Select transformation rule pattern:
  Procure *value* $object_1$ from $actor_2$ by providing value $object_3$ to $actor_2$ and offer value $object_1$ to $actor_1$ and receive value $object_2$ from $actor_1$
- Step2. Apply the pattern by replacing value model notions by actual instances:
  Procure *Information on Health care specialist clinics* (value $object_1$) from *Health care knowledge center* ($actor_2$) by providing *Registration* (value $object_3$) and offer *Information on Health care specialist clinics* (value $object_1$) to *Patient* ($actor_1$) and receive *Fee* (value $object_2$) from *Patient* ($actor_1$)

– Step 3. Actions:
  Introduce new Actor *Health care knowledge center* ($actor_2$).
  Add a new value exchange for procuring *Information on Health care specialist clinics* (value $object_1$) from *Health care knowledge center* ($actor_2$) to *Primary care physician.*
  Add new value exchange to provide *Registration* (value $object_3$) as compensation for *Information on Health care specialist clinics* (value $object_1$).
  Add new value exchange to offer *Information on Health care specialist clinics* (value $object_1$) to *Patient* ($actor_1$).

The actions in step 3 will lead to the introduction of a new actor - Health care knowledge center to procure the Information on Health care specialist clinics (see Figure 3). Thereby, there will be two new value exchanges between Primary care physician and the Health care knowledge center, for the procurement and for providing Registration as compensation. These new exchanges will be connected to the value activity Treatment and Referring to special care in the Primary physician. Also a new value exchange is added in an existing value interface between Primary physician and the Patient to transfer the Information on Health care specialist clinics.

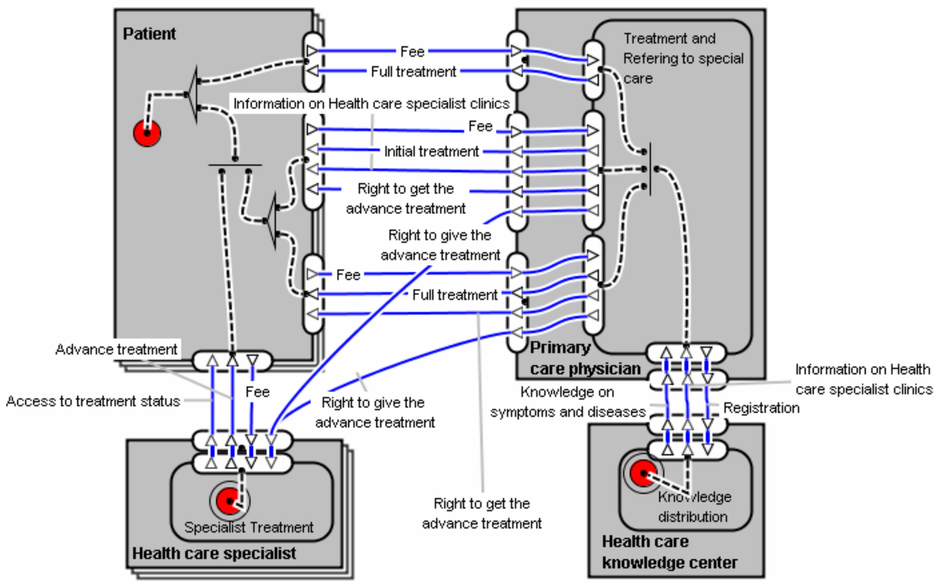

Fig. 3. Enhanced e^3 value model for the eye-care case.

5 Conclusion

This paper has addressed the problem of aligning business models with goal models. A method was proposed that takes as input a goal model and an as-is business model and transforms it into a new business model that conforms to the goal model. The link between goal and business models is primarily through the notion of means. The proposed approach offers a number of benefits:

– *Uniform goal model formulation*. Formulating goals and means in terms of business model concepts make goal models more uniform and objective in the sense that different designers will express a given goal or means in similar ways.
– *Flexibility and separation of concerns*. The approach allows for flexibility since the means definition distinguishes between strategy and tactics, i.e. how the strategy will be carried out. If the goal modeler is not aware of the tactics at design time of the goal model, that tactics-part may be left to be filled in by the value modeler instead. This also means that the tactics-level details of a goal model can be tailored to the situation at hand by selecting appropriate patterns.
– *Traceability*. It is possible to relate the components of a goal model to those of a business model, as the goal model has to be formulated in terms of the notions in the business model. Furthermore, components of a business model are directly motivated by the goal model.

A number of issues need to be addressed in future work. The main issue is the completeness of the means templates. The templates are currently confined to a small number of basic archetype activities, aligned with the modeling scope of the *e^{3}value*.

References

1. McCarthy W. E. 1982. The REA Accounting Model: A Generalized Framework for Accounting Systems in a Shared Data Environment. *The Accounting Review*.
2. Gordijn J., Akkermans J.M. and Vliet J.C. van. Business Modeling is not Process Modeling. *Conceptual Modeling for E-Business and the Web*, LNCS 1921, Springer-Verlag, pp. 40-51.
3. Osterwalder A. 2004. The Business Model Ontology. *A Proposition in a Design Science Approach. PhD-Thesis*. University of Lausanne.
4. Andersson B., Bergholtz M., Grégoire B., Johannesson P., Schmitt M., Zdravkovic J.: From Business to Process Models - a Chaining Methodology. Proceedings of the BUSITAL'06 Workshop, Luxembourg. Namur Univ. Press, 2006 pp. 211-218. ISBN: 978-2-87037-525-9
5. Johannesson P., Andersson B., Bergholtz M., Edirisuriya A. and Ilayperuma T. Zdravkovic J. On the Alignment of Goal Models and Business Models. Submitted to *REA 25:A Celebration of the REA Enterprise Ontology*. June 13-15, 2007, Newark, Delaware, USA.
6. Gordijn J., Petit M., Wieringa R. 2006. Understanding business strategies of networked value constellations using goal- and value modeling. In Martin Glinz and Robyn Lutz editors, *Proceedings of the 14th IEEE International Requirements Engineering Conference*, Pages 129-138, IEEE CS, Los Alamitos, CA, USA.
7. REMS (REMisslusS) Project, http://www.rems.se, last accessed 2006-06-29
8. Yu S. 1995. Models for supporting the redesign of organizational work. *Proceedings of the Conference on Organizational Computing Systems (COOCS 1995)*. Milpitas, California, USA. ACM press, pp 226-236.
9. Business Rules Group (BRG). Business Motivation Model (BMM). Available at http://www.businessrulesgroup.org/. Last accessed 14.2.2007
10. SWOT Analysis. Wikipedia, the free encyclopedia. Avalable at http://en.wikipedia.org/wiki/SWOT. Last accessed 14.2.2007
11. Hruby, P. Model-Driven Design Using Business Patterns. Springer Verlag. ISBN: 3-540-30154-2

e^3*competences* : Understanding core competences of organizations

Vincent Pijpers and Jaap Gordijn

Free University, FEW/Business Informatics, De Boelelaan 1083a, 1081 HV Amsterdam, The Netherlands. (v.pijpers, gordijn)@few.vu.nl.

Abstract. In this paper we present the e^3*competences* ontology, which enables us to conceptually model an organization's *core competences* such that we can(1) *identify* core competences and (2) analyze whether value activities positively or negatively *contribute* to the core competences of the organization at hand. The e^3*competences* ontology, which has an internal view on organization and is partially based on the e^3*value* ontology, is positioned next to the e^3*forces* ontology, which has an external view on organizations.

1 Introduction

As early as the 1980's the importance of information technology (IT) on an organization's business strategy has been stressed. Since then, IT has evolved from simple databases to worldwide service oriented architectures, making the impact of IT on an organization's business strategy in the present even more important [4].

In (traditional) business literature two distinctive, although complementary, views on business strategy can be distinguished. One view considers the *environment* of an organization to be the most important strategic motivator. This strategy school is grounded in the work of M. Porter [10]. Their understanding is that *forces* in the *environment* of an organization determine the strategy the organization should chose. An organization should position itself such that competitive advantage is achieved over the competition and threats from the environment are limited. In contradiction, the second school considers the *internal* competences as the prime motivator for an organizations business strategy. This school is rooted in the belief that an organization should focus on *unique resources* [1] or *core competences* of an organization [11]. Core competences are those activities with which an organization is capable of making solid profits [5]. For the continuity of the organization it is best to choose a strategy which focuses on the organization's core competences.

In previous work [8, 9] we focused on the "environmental" school of business strategy. In this paper however, we focus on the "competences" school of business strategy. The goal of this paper is to present an ontology, named e^3*competences* , which we will use to conceptually model and analyze the core competences of an organization. By looking at *internal* business strategy motivators e^3*competences* complements the *external* business strategy motivators considered by e^3*forces* [8, 9]. Together e^3*competences* and e^3*forces* draw up the conceptually modeling framework e^3*strategy*, which is intends to understand and analyze strategic business motivations of actors in a networked value constellation. As with the e^3*forces* ontology, we closely relate the e^3*competences* ontology to the e^3*value* ontology developed by Gordijn and Akkermans [2, 3], such that a well integrated set of business ontologies for networked value constellations emerges. Because the e^3*value* ontology, like the e^3*forces* ontology, focuses on the environment of organizations, it is necessary to complement the e^3*value* ontology with additional *internal* constructs. These additional constructs will make the e^3*competences* ontology suitable for analyzing the *core competences* of an organization. To present the e^3*competences* ontology and demonstrate its practical use we utilize a small desk-based case study to analyze two different situations: (1) The organization does *not* have clear understanding of what its core competences are, here we use e^3*competences* to determine the core competences. (2) The organization *has* identified its core competences, here we use e^3*competences* to determine to what extent the organization's value activities/transfers contribute to the core competences.

This paper is constructed as follows: first we introduce a desk based case study. Subsequently we present the constructs used in the e^3*competences* ontology. Next, we demonstrate how e^3*competences* is used to reason about the core competences of an organization. Finally, we reflect on extending the e^3*value* ontology for strategic analysis, present conclusions and make suggestions for further research.

2 Case Study

To present and demonstrate the e^3*competences* ontology we consider a constellation consisting of three organizations: (1)*Airport Inc.*, hereafter referred to as "AP", who owns and exploits a physical airport. (2) *Air Traffic Control*, hereafter referred to as "ATC", responsible for the air traffic management (ATM) (eg. landing and take off) at the airport. (3) *Dispatcher*, hereafter referred to as "DP", who is responsible for services such as loading and unloading of the planes. The constellation has two basic groups of customers: (1) *Airliners*, who acquire infrastructural services (eg. (un)loading) from "DP", air traffic management from "ATC" and infrastructural services (eg. a runway) from "AP". (2) *Passengers*, who acquire value objects from "AP" in the form of infrastructural services (eg. shops and other facilities).

Fig. 1 provides a basic e^3*value* model for the constellation (for more information on e^3*value* , see [2,3]). As can been seen, "AP", "ATC" and, "DP" exchange value objects to provide other value objects to "airliners" and "passengers". What however cannot be seen is which value activities are, or contribute to, the core competences of the various organizations in the constellation. As motivated earlier, this is an important component for understanding the strategy of an enterprise.

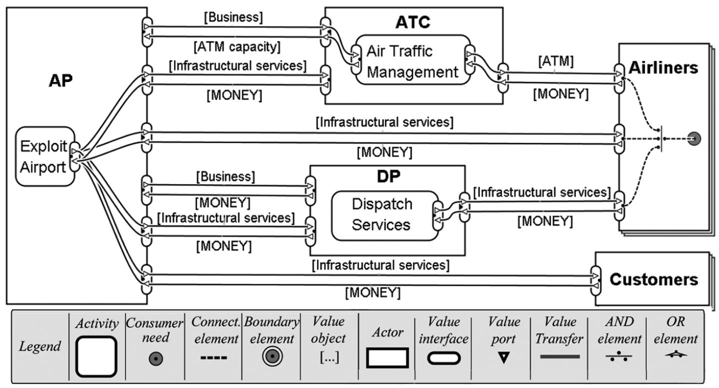

Fig. 1. Basic e^3*value* model

3 The e^3*competences* ontology

As with the e^3*forces* ontology [8, 9], we use the e^3*value* ontology [2, 3] as a base for the e^3*competences* ontology. In the e^3*value* ontology it is possible to model internal business activities of actors, but the e^3*value* ontology mainly - and intentionally - focuses on value transfers *between* organizations. In the e^3*value* ontology the construct "value activity" only intends to answer the modeling question "who does *what*, to create a profit" (as this is a design choice while developing constellations). Value activities are not used to understand internal working of actors. Furthermore, value activities can *only* be related through value objects they transfers, *decomposition* of value activities is for example not explicitly possible. Nor is it possible to distinguish between a "normal" value activity and a "core competence" value activity, making it impossible to identify which activities are key for the organization's business strategy. For these reasons we introduce the following concepts in the e^3*competences* ontology:

- *Core Competences.* The first additional construct is *core competence.* Core competences are: "activities that critically underpin an organization's competitive advantage; they create and sustain the ability to meet customers need better then the competition " [5]. Basically core competences are what makes an organization unique. Core competences will be modeled as rounded squares with an *extra bold line.*
- *Unique Resources.* Related to core competences are *unique resources.* To posses, or have access to, a unique resource is not sufficient to create competitive advantage [11]. Only if a unique resource is *adequately exploited* the activity of exploiting the unique resource will become a core competence. We consider unique resources in the broadest sense possible, unique resource can range from specific employees to access to natural resources. Furthermore, unique resources can either stem from the organization itself or can be acquired, via value transfers, from another organizations. We include "unique

resource" into the model to be able to show that if an organization has unique resources, the organization does, or does not, adequately exploit these resources and therefore has, or has no, core competences. Unique resources will be modeled as rounded squares with a *dotted line*.
- *Sub-value activity.* We adopt the value activity construct from the e^3*value* activity, but we want to be able to decompose this value activity into *sub-value activities*. We base our decomposition method on the decomposition of "tasks" into "sub-tasks" as done in *i** [12]. A higher level value activity can *only* be completed if *all* sub-value activities are completed. In addition, every sub-value activity belonging to one higher level value can be performed *independent* from the other sub-value activities. Sub-value activities will be modeled as rounded squares with *dashed lines* and are connected to value activities by a single value transfer.
- *Contributions.* Finally, we want to model *positive* or *negative contributions* of various value activities to core competences. It is our understanding that an organization can posses value activities which are not core competences, but who do, or do not, contribute to an organization's core competences. For example, air traffic management is the core competence of "ATC". Recruiting air traffic controller is not part of the core competence, but this value activity does positively contribute to the core competence. Would for instance "ATC" also have a web design value activity, then this value activity would not positively, thus negatively, contribute to "ATC" 's core competence. We model this by including *contribution arrows*, who are labeled with either a "+" or "-". Here we roughly follow *i** [12] .

3.1 Case 1: Identifying core competences

The first use of e^3*competences* is to identify the core competence(s) of an organization. We use a stepwise approach that will enable us to interrelate *unique resources* to *value activities*, which are connected to value transfers and thereby acquire or sell value objects. These relations enable us to identify the core competences of the organization. As a starting point we consider an actor, modeled in an e^3*value* model, for which we want to understand its core competences. Because in an e^3*value* model "value activities" are not intended for such analysis, we start with a clean sheet by removing all existing value activities and connection elements within the actor under investigation.

1. To enable us to consider the *complete* range of (sub-)value activities conducted to (1) acquire value objects (eg. resources) and (2) sell value objects, we connect *one* value activity to each of the *value transfers*.
2. We identify the (unique) *resources* an organization acquires from other organizations. If the organization has activities where a value object is *received*, other then money, we consider this to be a *resource* acquired by the organization and for now we replace the connected value activity with the (unique) resource acquired from the other organization.

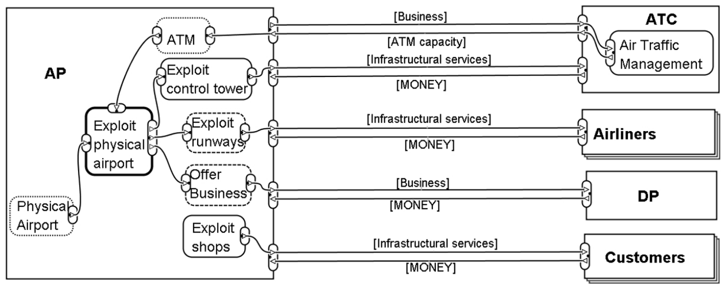

Fig. 2. e^3*competences* : Identifying core competences

3. We determine if value activities, modeled in step 1, have a common denominator; are a number of these activities actually *sub-value activities* leading together to a "higher level" value activity? If so, then these sub-value activities are modeled as such and connected to the higher level value activity. If not, then the value activities are left alone.
4. We link the value activities (identified in the previous step) to resources (identified in step 2), but only if the resource is needed by the value activity for its execution.
5. We determine if the value activities require other resources than those acquired from other organizations. In this step we try to identify the (unique) resources an organization has within its organization. If there are unique resources within the organization, then they will be included and connected to the value activities.
6. Next, for each of the resources modeled in the organization, we determine to what extent the resource is an *unique resource*. If an organization acquires a resources from another actor we question if other organizations (eg. competitors) have *access* to the same or a similar resource. If the organization posses the resource we question how likely it is that other organization possesses the same resource. If the resource is *not* an unique resource, the construct is removed from the model.
7. Value activities which are connected to *unique resources* and are connected (via sub-value activities) to value transfers are considered to be core competences. Value activities which are not connected to unique resources remain *value activities*.

Due to space considerations, we only consider "AP" in this case. Fig. 2 shows the e^3*competences* model for the first situation. Following the steps above we were able to identify that "AP" acquires one unique resource, "ATM", and possesses one, "Physical Airport". Furthermore, the individual value activities, connected to the various value transfers, are, except one, sub-value activities that lead to the value activity "Exploit Physical Airport". Since this value activity is connected to both unique resources, it is a *core competence*.

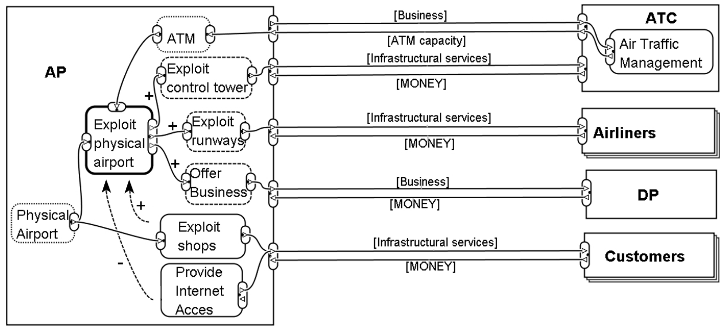

Fig. 3. e^3*competences* : Contributions to core competences

3.2 Case 2: Value activity contribution to core competences

The second use of e^3*competences* is to analyze if value activities positively or negatively contribute to the, earlier identified, core competence(s) of the organization. Again we use a stepwise approach. By analyzing the relationship between core competences/unique resources and value activities, we are able to determine positive or negative contributions of value activities. Again we start with a clean sheet by removing all existing value activities and connection elements within the actor under investigation.

1. Include the core competences (earlier identified) into the organization.
2. To enable us to consider the *complete* range of (sub-)value activities conducted to (1) acquire value objects (eg. resources) and (2) sell value objects, we connect *one* value activity to each of the *value transfers*. At this point we do *not* link the core competences to the individual activities.
3. We identify the *resources* acquired from other organizations. If the organization *receives* a value object, other then money, we consider this to be a resource *acquired* by the organization and for now we replace the connected value activity with the (unique) resource acquired from the other organization.
4. We determine which of the value activities, remaining from step 2, are sub-value activities of the core competences. Those that are, are modeled as such. We assume that sub-value activities have a positive contribution to the core competence, since the core competence can only be executed if all sub-value activities are executed.
5. Next we identify the (unique) resources an organization possesses, which are needed to execute the core competence and connect them accordingly. Most commonly the unique resources identified should not equal those acquired from other organizations via value transfers.
6. At this point we identify if the remaining value activities from step 3 contribute positively or negatively to the core competences. We use the following criteria: If the value activity *utilizes* one of the unique resources, then it positively contributes to the core competence. If the value activity does *not*

utilizes one of the unique resources, then it negatively contributes to the core competence. Which does not mean it is a "wrong" value activity, it just does not contribute to the core competence of the organization.

Fig. 3 provides the e^3*competences* model for this case. Again we only focus on "AP". The model shows that there are two value activities which are not sub-value activities: "Exploits Shops" and "Provide Internet Access". "Exploit Shops" does however utilize the unique resource "Physical Airport"; the shops are part of the physical airport. The value activity "Provide Internet Access" utilizes resources such as IP access, routers, etc. It does however not utilize the physical airport and therefore does not contribute to the core competence. According to business literature [5], "AP" should focus on its core competence and seize or outsource its "Provide Internet Access" activity.

3.3 Relevance for IS development

At first business strategy concepts such as core competence might seem distant from IT development. But the role of IT on developing and executing a business strategy is becoming more important [4]. Furthermore, understanding the *context* of IS is becoming increasingly important (eg. [12]). Models such as e^3*competences* , but also e^3*value* [2] and *i** [12], aid (chief) information officers to *explore* how the organization's IT/IS infrastructure (can) positively or negatively contribute to the organization's core competences and *design* the organization's IT infrastructure accordingly. For instance, deploying IT to acquire "ATM" from "ATC" faster/better might be a solid investment due to its unique nature and importance to the core competence of "AP". In addition, deploying IT to "sell" the core competence to buyers (eg. airliners) could increase the potential range of buyers. Such understanding and exploration of the organization on a business strategy level should aid in developing a better IT strategy and better business-IT alignment.

4 Related Work

The e^3*competences* ontology approach is related to "Enterprise Architecture" research. Enterprise architectures model and analyze organizations from five different perspectives [7], where the "Resource" view has most in common with e^3*competences* , since it also takes resources and capabilities into account. Resources and capabilities are however viewed from a process perspective, instead from a business strategy perspective. Furthermore, enterprise architecture are often complex and take many aspects of an organization into account [6], while the e^3*competences* ontology is lightweight and focuses on core competences only.

5 Conclusion

In this paper we have presented the e^3*competences* ontology, which has enables us to (1) identify the core competences of an organization and (2) analyze if value activities positively or negatively contribute to the core competences of an organization. The e^3*competences* ontology is an first attempt in better understanding organizations from a business "competences" perspective, yet IS developers could use an e^3*competences* model to *explore* how the organization's IT/IS infrastructure can positively contribute to the organization's core competences and *design* the organization's IT/IS infrastructure accordingly. In addition, we position the e^3*competences* ontology, which has an *internal* view on an organization's business strategy, next to the e^3*forces* ontology, which has an *external* view on an organization's business strategy. The combination of both ontologies (e^3*strategy*) enables us to fully analyze and understand the *strategic motivations* of an organization participating in a networked value constellation. Further research is however needed to examine and conceptualize the exact relationship between an e^3*competences* model and an e^3*forces* model.

Acknowledgments This work has been partly sponsored by NWO project COOP 600.065.120.24N16.

References

1. J. Barney. The resource-based theory of the firm. *Organization Science*, 7(5):131–136, 1994.
2. J. Gordijn and H. Akkermans. E3-value: Design and evaluation of e-business models. *IEEE Intelligent Systems*, 16(4):11–17, 2001.
3. J. Gordijn and H. Akkermans. Value based requirements engineering: Exploring innovative e-commerce idea. *Requirements Engineering Journal*, 8(2):114–134, 2003.
4. G. Hidding. Sustaining strategic advantage in the information age. In *Proceedings of the 32nd Hawaii International Conference on System Sciences*. IEEE, 1999.
5. G. Johnson and K. Scholes. *Exploring Corporate Strategy*. Pearson Education Limited, Edinburgh, UK, 2002.
6. G. Khoury, S. Simoff, and J. Debenham. Modeling enterprise architectures: An approach based on linking metaphors and ontologies. In *Proceedings of the 2005 Australasian Ontology Workshop*, volume 58, pages 41–46, Sydney, AU, 2005.
7. D. Liles and A. Presley. Enterprise modeling within an enterprise engineering framework. In J. Charles, D. Brunner, and J. Swain, editors, *Proceedings of the 1996 Winter Simulation Conference*, pages 993–999, San Diego, CA, 1996.
8. V. Pijpers and J. Gordijn. Does your role in a networked value constellation match your business strategy - a model based approach. Accepted at Bled eConference, 2007.
9. V. Pijpers and J. Gordijn. e3forces: Understanding strategies of networked e3value constellation by analyzing environmental forces. 2007. Accepted at CAISE 2007.
10. M. E. Porter. *Competetive Strategy. Techniques for analyzing industries and competitors*. The Free Press, New York, NY, 1980.
11. C. K. Prahalad and G. Hamel. The core competence of the organization. *Harvard Business Review*, 68(3):77–93, May/June 1990.
12. E. Yu and J. Mylopoulos. An actor dependency model of organizational work - with application to business process reengineering. In *Proceedings of the conference on Organizational computing systems*, pages 258–268, New York, NY, 1993. ACM Press.

Comparative analysis of process and value perspectives for insight in business cooperation

Wei Feng, Jon Atle Gulla and Darijus Strasunskas[1]

Dept. of Computer and Information Science,
Norwegian University of Science and Technology, NO-7491 Trondheim, Norway
{weif, jag, dstrasun}@idi.ntnu.no

Abstract. In this paper we exercise a combination of two modelling techniques investigating how they assist in analysis of business processes. For this purpose, we compare e3value and UML Activity models in a case study against a set of business process analysis aspects. The paper concludes that none of the models is alone sufficient for the purpose. However, the e3value model better covers the required analysis aspects and serves as a good basis for further extensions.

1 Introduction

Business cooperation is these days critical for business success, as better collaboration opens up for new business opportunities. Companies join into business networks in order to reduce costs and enhance their competitive strength. In order to optimize collaboration in such a setting, both process and value interactions should be optimized among participating nodes.

Business optimization may increase the company's productivity or profitability. Nevertheless inadequate optimization may also cause negative effects. Before an organization changes its business processes, it is necessary to carry out thorough qualitative or quantitative evaluations of the options available.

In this paper we investigate the value and process perspectives in enterprise modelling. First, we aggregate a list of critical aspects according to which business processes need to be analysed. Then we use the e3value technique [2] to model value related aspects and UML 2 Activity diagram to model the process part. An industrial case study is used to create the models. The analysis is conducted reflecting on both, the modelling process and the models.

2 Motivation and Related Work

There is a need for holistic enterprise modelling technique to better understand and align business processes and information flows. Already in 1978, Zachman [13] defined information system architecture as "the sum total of all information-related

[1] Authors' names are listed alphabetically.

flows, structures, functions and so on, both manual and automated, which are in place and/or required to support the relationships between the entities that make up the business." The idea is to get a complete picture of the information technology used in an organization by analyzing and modeling different aspects. Later the term enterprise architecture was coined, a.k.a enterprise information architecture (EIA). The purpose of enterprise information architecture "is to align the implementation of technology to the company's business strategy" and "to make technology serve innovation economics" [1]. Enterprise modeling is considered an important technique for IT and business alignment.

The challenge of enterprise modelling was approached by developing families of modelling languages integrating various levels of business management. For instance, Gustas and Gustiene [4] claim three levels of information system models are necessary for maintenance of systematic change, e.g., in order to understand why a technical system component is useful and how it fits into the overall organisational system. These levels are as follows: pragmatic level, semantic level and syntactic level. However, traditional information systems modelling techniques were not adequate to analyse the value EIA might provide.

Hammer [5] defines a business process as a "group of tasks that together create a result of value". Not much has been done in the area of relating business processes with business goals and value perspectives in modelling. There is however some recent work analysing the notion of value object [12]. Also, there is ongoing research relating value models with goal models [3], annotating process models with goals [8], extending UML Activity diagram with goals and performance measures [7] or business intelligence objects [11]. Hessellund [6] proposes to extend the popular REA model [9] with Location and Transport entities to be used for supply chain modelling. Originally the REA model was introduced to model accounting phenomena in enterprise information systems [9] and it consists of three basic entities: resources, events and actors.

3 Framework for Analysis

Business process models and value-centric models have a different focus. Whereas process models emphasize control sequences of activities, messages, data and objects, value models focus on the interchange of value objects without detailed analysis on what activities are necessary in order to exchange the value objects.

We have in our research identified a set of features that are important for the analysis of business alliances, their interaction and value exchange. These features are used to analyse value models and process models in order to identify strong and weak aspect of these two modelling techniques. An important objective of our work is to identify commonalities between the techniques and investigate how they can be combined to yield a more complete analysis of the business. The features are as follows.

F1. Stakeholders and their role. This aspect is important in order to identify all possible stakeholders that are directly or indirectly involved in a business process in question. This would allow analysing who starts the process, who carries it out and who terminates it. A role-based process modelling language would fulfil this need.

F2. Business goals. We need to be able to relate activities to business goals or assign intermediate goals for particular activities in order to investigate how they contribute to overall business goals. This would provide the material needed for managers to communicate better in an organization.

F3. Business activities. We need to have an overview of all activities in a business process. This is important for refining and optimizing the process.

F4. Process control. We need to model how a process is/could be executed. Here actions would identify how many operations are necessary to perform an activity. This would allow us to analyse the sequence of actions, even schedule them.

F5. Object properties and status. We need to represent rich information about an object used in a particular activity, whether this object serves as a tool to perform an activity or is the purpose for the activity.

F6. Value-adding activities. This feature is necessary in order to differentiate between supportive activities and value-adding activities. It would help us in analysing where and what values are created and transformed in which parts of the network.

F7. Quantification of value exchange. We need a method to quantify value exchanges. Here we are interested in assigning economic values to exchanged value objects, which would allow assessing an overall profitability performance of the activity, evaluating future prospects and success of new business opportunities.

F8. Performance metrics. Associated key performance indicators would help us optimize execution times, costs and value of business processes.

For the analysis of value and process perspectives, we use the e3value [2] value modelling technique and UML 2 Activity diagram. These two techniques are dominant in their perspective and modelling tools are easily available[2].

4 Case Study

Our case study is based on the Norwegian agricultural sector, in general, focusing on activities around the Felleskjøpet[3] company. In the agricultural sector there are many stakeholders, including (but not limited to) Felleskjøpet, farmers, slaughters/ processors, retailers, consumers, the government, media and accounting firms, banks, etc. In order to provide an explicit and relatively uncomplicated model representation, we limit the scope to five main stakeholders, such as, Felleskjøpet, Farmers, Slaughters/Processors, Retailers and Consumers. Furthermore, some processes are merged to represent a relatively general and common business process. For instance, we omit the business process that farmers sell meat or plant directly to consumers in local markets.

Figures 1 and 2 below exemplify value and process models, in e3value and UML 2 Activity diagram, correspondingly. Next we describe the case and discuss the desired modelling properties in more details.

Farmers. Felleskjøpet business model is structured around providing services to its members, i.e. farmers. As mentioned above, we exclude the government though it is a

[2] See http://www.e3value.com and http://argouml.tigris.org/, respectively.

[3] The Norwegian agricultural purchasing and marketing co-operation, http://www.felleskjopet.no/english

major stakeholder here providing not only quotas for production by farmers, but also paying various subsidies, issuing laws, etc. However, there are three major value exchanges in cattle production. They are about: 1) equipment needs, 2) grass needs/ grass provision, 3) feed needs/grain provision.

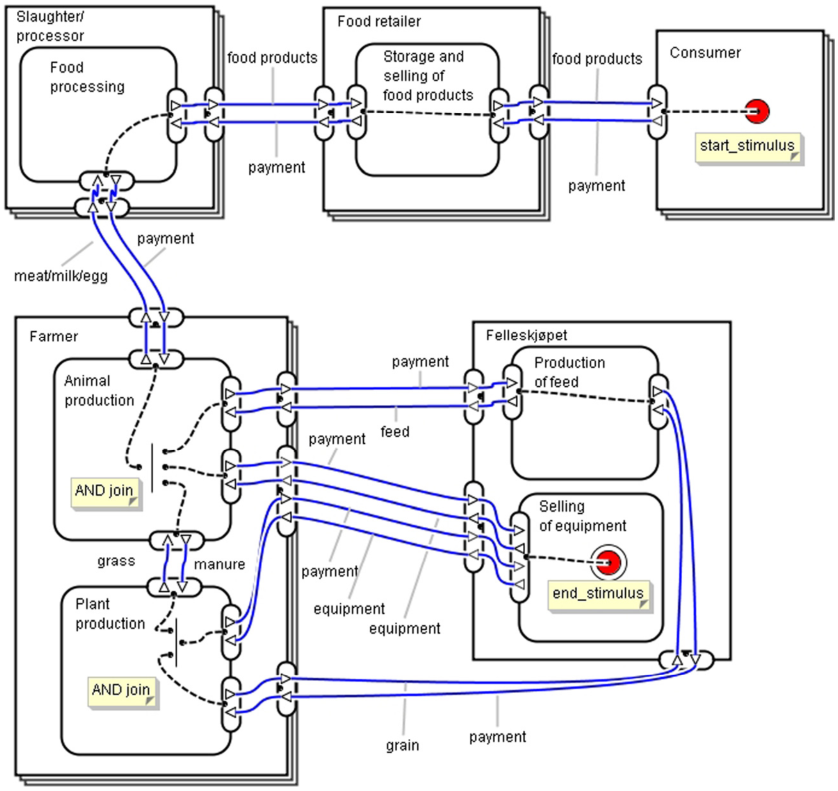

Fig. 1. High level value model of agricultural sector (using the e3value modelling technique and tool).

- Equipment needs. For cattle/plant production, farmers have to buy equipments such as tractors or batchers from suppliers. In return, farmers pay them. In this case, we take Felleskjøpet as the only supplier for equipments.
- Feed needs/grain provision. In stock-raising, farmers need to provide feed to animals. In this scenario they buy feed from Felleskjøpet and pay it in return. Grain producers sell their harvest to Felleskjøpet, where the animal feed is produced.
- Grass needs/grass provision. Stock-raising requires grass to feed live stocks such as cows and sheep. In the cattle farming, farmers can get milk and therefore it is a value added activity. Farmers obtain grass from other farmers or produce them by themselves.

Cattle farming also yields manure which farmers can utilize in their plant production or offer to other farmers. The value exchange normally happens between two different actors, but here the special thing is that this exchange happens between two identical actors. Farmers are grouped into a market segment used to show a set of actors, so this exchange can be shown between two activities inside one market segment. In total there are about 50,000 farmers in Norway.

Felleskjøpet. There are two value activities relating Felleskjøpet in this model. One is selling of equipment, and another is the production of feed. The latter includes two value exchanges - grain needs and feed provision. The former is the main activity of Felleskjøpet, since farmers need equipment for both animal production and plant production. In addition to Felleskjøpet, there are 2-3 other companies that serve these roles.

Slaughters/processors. For simplification we represent these two actors as one. The core business of these two actors is to process meat/milk/eggs and produce food products for end-consumers. There are in Norway 2-3 major companies in this market, and they are owned by the farmers.

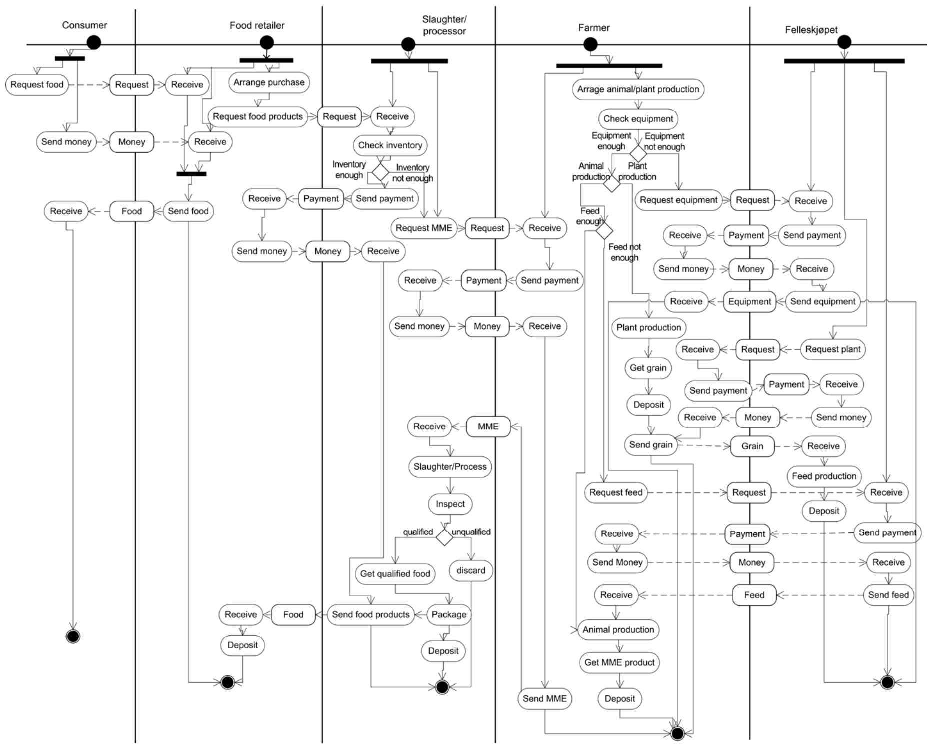

Fig. 2. An excerpt of process model (UML Activity diagram)

Food retailers. Food retailers deliver food to local stores and shopping malls. Their activity is centred around two value adding activities, namely, storage and retail of food products. The retailers are dominated by a few large chains that have traditionally been substantially more profitable than the other actors in the value chain. Finally, *consumers* buy food at a local grocery store.

5 Comparative analysis of process and value models

An analysis of the models with respect to the earlier defined features reveals some interesting differences between process modelling and value modelling.

F1. Stakeholders can be well represented using both modelling techniques. In a process model they are usually included only if they are actors or participants in a particular activity or process, while a value model is more focused on stakeholders that not necessarily are participants of the activity. For instance, for a state or a nation, a national agriculture is a valuable economic activity despite the direct costs associated with it, as it provides emergency preparedness in case of war. Therefore, we claim that a value model is better designated for modelling benefits of various stakeholders, even those not directly involved in a particular activity.

F2. When it comes to relating activities to business strategies and goals, both techniques lack adequate modelling constructs. However, if considering the optimal value balance among all involved stakeholders or a particular stakeholder being an overall goal, then certainly this can be exercised using the value modelling technique.

F3. Business activities are modelled in both perspectives, though value models are designated to analyse the value-adding activities and therefore do not have any means to model other activities. However, decomposition of value activities to their constituents or supportive activities would allow for optimization of the value activities, which in return would be more efficient and provide a bigger value.

F4. The activity diagram has all required constructs to denote and model activity execution sequence, schedule it and represent possible alternative path for execution of a particular activity after a certain decision. However, not all these characteristics are present in a value model. For instance, figure 1 only indicates which value object consumers exchange with retailers. It indicates neither how many actions consumer has to perform nor in what sequence these actions are typically performed.

F5. Any of the two techniques does not provide appropriate constructs for representing object properties. Both analysed modelling techniques allow for an implicit status change tracking. For instance, farmers are delivering cattle to a slaughter/processor prior to a veterinary check, i.e. kind of not inspected meat. Having modelled more detail processes, meat quality inspection would be one of the activities at slaughter site, potentially represented in both models. Successful execution of this activity would implicitly suggest change of meat status. However, such implicit support is not enough to claim that this feature is supported.

F6. This feature is about being able to explicitly model value adding activities. A certain winner here is the value model.

F7. Here we are interested in assigning economic values to exchanged value objects. Having assigned values we can simulate overall profitability performance based on an estimated number of value transfers. For example, in figure 1, food retailers request food products from slaughters/processors, and offer payment for this. It represents that food retailers and slaughters/processors have economic reciprocity by the exchange. However, an activity model does not have ability to model economic reciprocity directly. Figure 2 cannot show this rule obviously. Furthermore, the e3value modelling technique allows aggregating all exchanges of value objects and computes a net present value figures for each of the involved stakeholders, where a positive net value flow would indicate an economic sustainability [10].

F8. Here we are interested in the possibility to model and analyse key performance indicators in order to optimize value transfers. This concerns not only value-adding aspects of activities, but also improving efficiency and execution time of activities. The value model assists in analyzing the economical benefits from executing an activity, however it has no means to analyse how this benefit can be increased by eliminating inefficient performance characteristics. Neither does an activity diagram, which in fact provides even less support for this purpose.

Table 1. Summarizing comparison of value and activity models

Feature ID	Value Model	Activity Model
F1. Stakeholders and their role	High	Medium
F2. Business goals	Medium	Low
F3. Business activities	Medium	High
F4. Process control	Low	High
F5. Object properties and status	Low	Low
F6. Value-adding activities	High	Low
F7. Quantification of value exchange	High	Low
F8. Performance metrics	Medium	Low

Table 1 summarizes the above discussion by assigning values to each of the features based on how well they are supported by value and activity models. In summary, the redesigning of information systems should simplify current operational processes, and achieve the goal of eliminating operational inefficiencies. Some operational processes which can not directly or indirectly contribute to profit adding might be omitted. Process models typically serve as guidelines for activity execution, facilitate stakeholders to examine potential pitfalls in a new business process. Comparison of two models gives better understanding on how business functions. But modelling two different perspectives is a labour-consuming and erroneous process. Consequently, there are two ways to go. The first way is creating a methodology that allows transformation of one model to another [10]. However, in this case it is questionable which model is more intuitive to be produced first, i.e. which of them possesses more relevant information for smoother transformation. The second way would be extending one of the modelling techniques by integrating required modelling constructs, e.g. [3, 7, 11].

6 Conclusions and Future Work

In this paper two modelling techniques have been applied on a real business case and compared. The value modelling perspective deals with the question “who is offering what (value objects) to whom”, and gets what in return. Activity models present how activities should be carried out. The models are analysed with respect to an aggregated list of features required for detailed analysis in business process optimization or change projects. We conclude that none of the models is sufficient alone for the purpose. However, the e3value model better covers demand for required information and serves as a good basis for further extensions.

In summary, there are two advantages of a value model: 1) it represents the economic value perspective in a model-based way; 2) it contributes to a better understanding of value-adding activities in a business process. The activity model provides two advantages: 1) it gives a reasonably detailed view on current operational scenarios; 2) it provides information needed to improve productivity by eliminating unnecessary operational actions.

Therefore, one of the future works is more tight integration of value and process models by introducing the lacking constructs. Further, we need to investigate how value/process models could assist in an economical assessment and financial comparison of alternative business processes, including key performance indicators for optimization of activities.

References

1. Chorafas, D.N. *Enterprise Architecture: For New Generation Information Systems*. Crc Press Llc, 2002.
2. Gordijn, J. and Akkermans, J.M. Value-based Requirements Engineering: Exploring Innovative e-Commerce Ideas. *Requirements Engineering* 8, 2003, 114-134.
3. Gordijn, J., Yu, E. and Raadt, B. van der. E-Service Design Using i* and e3value Modeling. *IEEE Software* 23(3), May 2006, 26-33.
4. Gustas, R., and Gustiene, P. Towards the Enterprise engineering approach for Information system modelling across organisational and technical boundaries. *Proc. of the 5th Intl. Conf. on Enterprise Information Systems, vol. 3*, 2003, 77-88.
5. Hammer, M. Beyond Reengineering – How the process-centered organization is changing our work and our lives. Harper Collins Publishers, 1996.
6. Hessellund, A. Supply Chain Modeling with REA. Technical Report, TR-2006-80. IT University of Copenhagen. January 2006.
7. Korherr, B. and List, B. Extending the UML 2 Activity Diagram with Business Process Goals and Performance Measures and the Mapping to BPEL. *ER Workshops 2006*, LNCS 4231, 2006, 7-18.
8. Lin, Y. and Solvberg, A. Goal Annotation of Process Models for Semantic Enrichment of Process Knowledge. *Proc. of the CAiSE 2007*, Springer-Verlag, 2007.
9. McCarthy, W.E. The REA Accounting Model: A Generalized Framework for Accounting Systems in a Shared Data Environment. *The Accounting Review* LVII(2), 1982, 554-578.
10. Pijpers, V. and Gordijn, J. Bridging Business Value Models and Process Models in Aviation Value Webs via Possession Rights. *Proc. of the 40th Hawaii Int'l Conf. on System Sciences* (HICSS'07), 2007.
11. Stefanov, V., List, B. and Korherr, B. Extending UML 2 Activity Diagrams with Business Intelligence Objects. *Proc. of the 7th Int'l Conf. on Data Warehousing and Knowledge Discovery* (DaWaK 2005), LNCS 3589, Springer-Verlag, 2005, 53-63.
12. Weigand, H., Johannesson, P., Andersson, B., Bergholtz, M., Edirisuriya, A., and Ilayperuma, T. On the Notion of Value Object. *Proc. of the CAiSE 2006*, LNCS 4001, Springer-Verlag, 2006, 321-335.
13. Zachman, J.A. A framework for information systems architecture. *IBM Systems Journal* 26(3), 1987, 276-292.

The co-evolution of business goals and business processes in a changing situation

Fériel Daoudi

Centre de Recherche en Informatique,
Université Paris1-Sorbonne
90 rue Tolbiac, 75013 Paris

feriel.daoudi@malix.univ-paris1.fr

Abstract: the ever-transforming economic environment incites organizations to adapt their business goals continuously. Business processes need to evolve in concert with those strategic changes. In order to maintain "business goals/business processes" adequacy, there is an approach that help analysts to conceptualize their future business model.

1 Introduction:

The frequent changes of the business environment become the main concern of an organization seeing that they affect its costs and benefits. The enterprise should adapt its behavior to the new contexts every time it is necessary. The enterprise's identity is represented by its business goals, its business processes and its information system [7]. The necessity of adaptation is evaluated by managers and stockholders. It depends on criteria such as duration of changes (temporary, permanent), kind of change (legal, geographic, etc), necessary investment, etc that will not be presented in this paper. If adaptation is decided, the tree cited levels are affected. The technical answers for this issue are represented by the panoply of software tools as enterprise resources planning (ERP), Enterprise Application Integration (EAI),etc. However, managers still need support approaches to design the business change before implementing it by an information system. In deed, managers cannot invest in the cited tool and IT evolution projects without understanding the scope/spread of changes in their organization. The adaptation of the organization is materialized by setting new strategic goals integrating the changing parameters. The To-be strategic goals can be derived by two manners: from scratch, inspired from the As-is goals. In both situations, designers must reverberate the changes on the supporting business processes. The repercussion of change aims to maintain or to create alignment between the two levels: strategic level and business level [7] [8]. However, in practice the derivation of the To-Be business process model is scarcely methodic in organizations. People skip this step because they do not understand the reasons to formalize/model what they perceive as a banality. The absence of formalization (traceability) leads to a break between managerial decisions and business changes. This can harm to the alignment between different levels of the organization. To avoid this situation, designers need mapping rules that allows the evolution of business processes in concert with organization goals in a straightforward way. This evolution of business processes is due to the changes occurred on either goals or the correction of an existent misalignment between the strategic level and the business one. The proposed approach support business staff in this task. In deed, the use of typologies of goals and business processes ensure the traceability between goals' changes and business ones. The typology of goals developed during the European project ELEKTRA inspired the proposed approach. This typology expresses the nature of change expected by the organization at the strategic level to drift from the As-Is situation to the To-Be situation. This typology was exploited in this paper and was completed by business processes typology in order to make possible the propagation of strategic change at the business level. The propagation is realized in two steps: listing the necessary changes on business processes and realizing them. Thus for each kind of change at the business level there is definite ways to enact it. Those ways are presented as impact enactment strategies. All those steps/concepts are detailed in the following sections and exemplified. The section 2 will present the goals typology. Then, section 3 will describe the business processes typology as well as the strategies to enact business processes changes. Finally, section 4 will conclude this work and presents the potential research directions.

2 Goals typology:

Because goals are the steering wheel of the companies, it is necessary to understand and to classify them. A goal is an target that an organization would reach or have reached. Organizational goals are materialized by business processes [7]. Although the goals expressed by managers have not the same granularity. This is the first dilemma that enterprise models' designers face: is any expressed goal linkable to a business process?. The second issue is the information expressed by goals. In deed, managers can express the same goal differently. It can also express intentions as well as reality. This formulation influences positively or negatively the solution that a designer can propose to ensure adequacy between business goals. Thus, a typology of goals can ease the mission of the creation or the maintenance of fitness. It will facilitate the selection of the right kind of goals when they evaluate the alignment in their organization or when they want to maintain it in a changing situation. Those two issues raised in the European project named ELEKTRA [4]. The aim of this project was to help a Greek electrical company to introduce changes in its practices by providing a method to represent the organizational knowledge. The proposed method is named EKD-CMM [4]. Concerning the granularity issue, EKD-CMM distinguishes between "*operational goals*" and "*non operational goals*". Operational goals are those that can be enacted directly by a business process whereas non-operational goals are those that have a high level of granularity/abstraction and cannot be linked straightforwardly to a business process. A non-operational goal is composed of various operational goals; it must be refined into operational goals in order to establish direct links with the business processes supporting them. The connectors used to represent the hierarchy of goals are discussed in [4].Other works use the notion of goal refinement in the alignment issue [2][5] as well as in the conception of information systems [1][6]. The refinement operation concerns the strategic goals. Each strategic goal is refined into sub-goals that can be either operational goal or non-operational goals. This operation is repeated until deriving completely operational goals that can be represented directly by business processes. As said previously, the information encapsulated in a goal influences the way to exploit it. A goal describes usually the targets that an organization wishes to reach in a stable state of her life cycle. However, some goals of managers state the wished changes in comparison to the actual situation of the organization. This observation shows how it is interesting for organizations to get describing goals when they are in a changing stage. It allows knowing the departure position (present situation), the arrival position (future situation) and above all the way to reach this last. Such a distinction appears in EKD-CMM [3] where are cited: the "*usual organizational goal*" and the "*change goal*". As defined above, an organizational goal describes the targets than an organization wants to realise. In opposite, a change goal is a way to describe the changes to perform on the current goals (present situation) in order to drift to the new aims of the organization. The change goal is composed of two parts: the present organizational goal and the impact type. The present organizational goal represents the targets than the organization want to realize in the As-Is situation. The impact type represents the changes planned on the As-Is goal in order to realize (reach) the To-Be situation. This typology is illustrated by the figure1 that shows the refinement of a strategic goal non-operational goal into operational goals. The approach [3] [4] presents five kinds of possible impacts: introduce, improve, stop, maintain, extend. Only four of them were kept to propose adequacy mechanisms with To-Be business processes of the organization. In deed, the "extend" impact type is presented as an extension of the scope of a given goal. Thus, this last will produce a new result different from the former. The finality of an extended goal is completely different from the initial one. This is similar to the introduction of a new goal seeing that the delivered product is changed. Seeing that the two cited impact types are overlapping, the set of impact types adopted in this paper was limited to the four impact types described below. The appliance of those impact type on organizational goals is illustrated in figure1 that shows the transition from the As-Is situation to the To-Be situation.

- The "Introduce" impact means that there is a new goal related to a new requirement of the organization that was not satisfied by the actual business processes of the enterprise. Example: Introduce (provide after sales service)

- The "Improve" impact means that there is an organizational goal that will be remained in the future goals of the organization but should be performed in a better way. Example: Improve (produce clothes for hiking)

- The "stop" impact type means that a current organizational goal will be stopped because it does not satisfy the future orientations adopted by an organization. Example: stop (rent cars to persons)

- The "Maintain" type means that the current organizational impacted goal will appear at the same form in the future situation of the organization. Example: maintain (rent cars to companies)

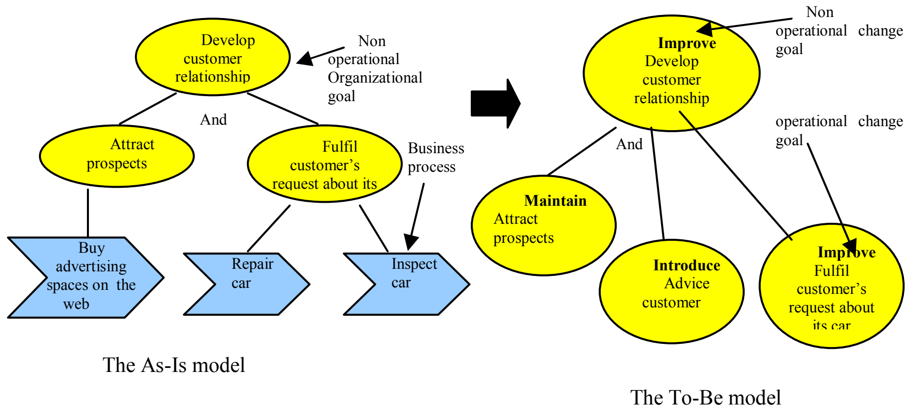

Fig. 1. The transition from the As-Is to the To-Be strategic level

3 Propagation mechanisms

The evaluation of the adequacy between goals and business processes concerns steady situation as well as a changing situation. An organization does not need to change its strategy to check the adequacy between its goals and the business processes supporting them. It is a way to verify that internal or external changes have not influenced discreetly its way of functioning. At the other hand, the organizations changing their orientations need to keep or to correct the alignment between its different levels [8] [7]. To facilitate this mission, the proposed approach uses the operational goals to establish the link between the strategic and the business level [4]. As said previously, operational goals represent the leaves in the hierarchy of goals of a given organization. This permits to get a straight link between the goal and the business processes supporting it. Thus in a changing situation, change operational goals replace the organizational operational goals in the connection of the strategic level with the business one. Indeed, they include managers' requirements about the evolution of their organization by referring to "the As-Is" situation that motivates their decision. Seeing that all the mutations expressed by change goals must propagated at the business level; different actions to perform on business processes are necessary.

The four types of impacts presented in section2: introduce, improve, stop, maintain represent completely the possible axis of change at the business level. The use of this impact typology at the business level leads to the distinction between "*stable business processes*" and "*changing business processes*". A stable business process is a set of linked activities that create value by transforming an input into a more valuable output. In opposite, a changing business process is a way to describe the changes to perform on an As-Is business process (stable business process) in order to fulfil the changes expressed by an organization. A changing business process is the appliance of an impact type on a stable business process. The kind of impact types to apply on As-Is business processes depends on managerial decisions derived from the organizational context that are not described in this work.The enactment of the change goals on the business level is realized when each changing business process is transformed into a stable business process. The way to realize the changes expressed by the changing business processes must be selected before executing the mutation itself. Indeed the impact type composing a changing business process can be enacted through different strategies. Once the strategy is selected, designers can perform the business changes that fulfill the strategic orientations of the organization as it is illustrated in figure 2. The choice of the strategy is influenced by the business knowledge as well as the organizational context (environmental constraints, needs, budget, time, etc) that will not be explored in this paper. The list of the possible strategies to enact to each impact type is detailed below.

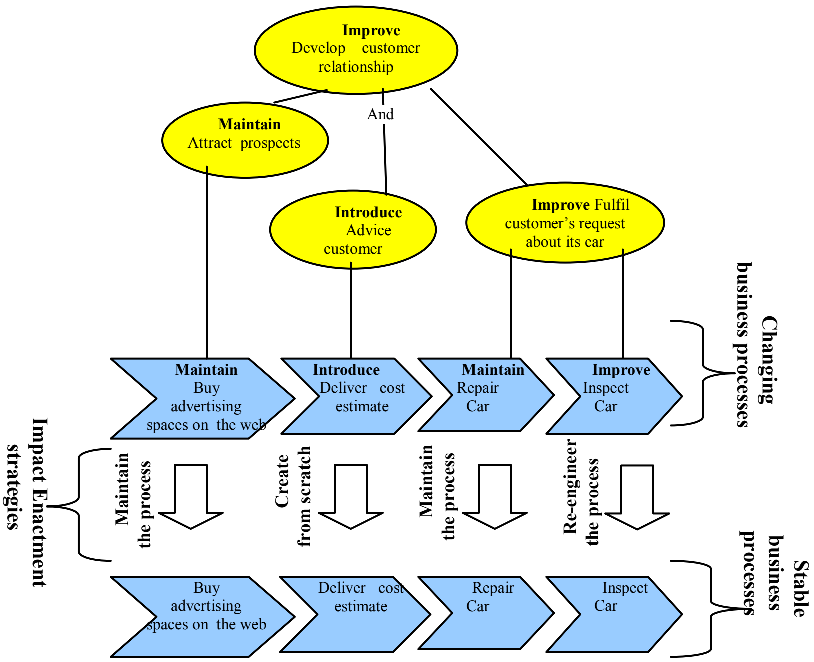

Fig. 2. Propagation of change on the business level

3.1 The "stop impact" enactment strategies

The presence of this impact means that an existing business process linked to an As-Is operational goal will be stopped because it does not satisfy the requirements represented by the change goal to which is linked. Thus, the organization has two alternatives: **stop the process**, **submit the process for the reuse**.

- **Stop the process:** the aimless process is stopped. This action can be performed by different ways. In deed, the organization can decide to stop immediately the process or to make this action progressively. This manner is influenced by the organisational context (size of organization, culture of the organization, cost, time, etc). This solution is adopted if the business process will not be useful to support another future goal. It can be also adopted if it fits better the economic situation of the enterprise than to submit the process as a candidate reusable process.
- **Submit the process for the reuse:** the aimless process is kept as a candidate process. This last can be reused and re-engineered to satisfy requirements that are not covered by other business processes. The submitted business process will be used if it provides elements necessary to the realization of a new requirements appeared in the To-Be strategy of the organization. It can also be exploited for the improvement of a given business process if it fulfils the required changes partially or in totality.

3.2 The “introduce impact” enactment strategies

The presence of this impact means that a new business process that did not exist previously is integrated to the set of business processes of the organization. Thus the organization has three alternatives: **create a process from scratch**, **re-engineer a submitted process**

- **Create a process from scratch:** a new process is created to support the new requirements submitted by managers. Creating a process from scratch occurs when there is no submitted business process (process proposed for reuse) inside the organization that can be reengineered in order to deliver the result expected by the new business process. This alternative occurs also when existing submitted processes misfit the orientations required by the new business process. Finally it can reflect a managerial choice when the creation of a new process is faster or less expensive to implement than the re-engineering of a submitted process.
- **Re-engineer a submitted process:** a process in quarantine (previously satisfying the As-Is strategy and that must be stopped) is selected to be re-engineered in order to support the To-Be strategy. The re-engineering is possible when the selected process can match the new aims of the organization after the introduction of changes. Those changes consist in completing the submitted process in order to deliver the desired result. It can also consist in changing radically the process and reusing a chunk of it in the building of the expected process. This choice is usually done when the reuse is less expensive and less time consuming than the creation from scratch.

3.3 The "improve impact" enactment strategies

The presence of this impact means that the organization intends to keep the same orientation but introduces ameliorations in the way of performing the impacted business process. Two alternatives exist to perform this operation: **re-engineer the initial process**, **re-engineer a submitted process and stop the initial process**.

- **Re-engineer the initial process:** the impacted business process is improved in order to satisfy the new parameters of requirements of the organization**.** The improvements concerns the way of functioning (activities, actors, resources, etc) of the business process. Those changes aim to ameliorate the performance of the process. They can also aim to integrate new adopted practices that have no relationship with the performance of the process (ISO norms, practices that allow tax deduction, etc).
- **Re-engineer a submitted process and submit the initial process to reuse:** the required improvements are not performed on the initial business process because of several factors such as: cost, competences, resources, etc. Thus, the designers should select amidst the submitted processes for reuse, a process that can match the expected ameliorations or can include them. If there is no candidate process that can undergo the required changes this alternative cannot be applied. As in the previous alternative, the changes concern the integration of new practices that have no relationship with the performance of the process (ISO norms, practices that allow tax deduction, etc). However, the introduced changes can aim also to ameliorate the performance of the process.

3.4 The "maintain impact" enactment strategies

The presence of this impact means that the business processes satisfies the To-Be change operational goal linked to it, thus it will be kept. This means that it will produce the same result. Two alternatives are possible to perform this operation: **maintain the process**, **maintain of the process.**

- **Maintain the process:** the business process to maintain is not altered by any change. This usually happens when the performance of the process is satisfying. It happens also when there is no context's change that can affect the progress of the concerned process.
- **Maintain the process with adding framed changes:** though the business process to maintain should deliver the same result, the way to perform it varies. The result is the final product/value delivered by the process. Thus, the allowed changes are those that do not alter provided by the business process. All the concepts participating in a business process (actors, activities, resources, etc) are concerned by those improvements. The improvements are due to internal or external changes that alter the functioning of the process and that are unavoidable (new legal regulations, appliance of new norms, retirements, etc).

4 Conclusion

In this paper, we addressed the problem of how to repercute the changes of the To-be strategic level on the business process level. An approach is proposed to support designers in the repercution of goals' change on business processes. It is based on goals and business processes typologies. The mapping between the two levels is represented by the link between operational goals and business processes. The approach offers landmarks to designers in order to express organization changes. There are change goals that indicate the required changes at the strategic level. There are also changing business processes that represent the expected changes at the business level in order to propagate the strategic requirements of the organization. Finally, there are strategies to enact change types at the business level. The appliance of those strategies ends the change step by transforming the changing business processes into stable ones. Referring to the As-Is situation for the enactment of change helps business and managerial staff in the improvement task of their organization. The use of the As-Is as an improvement reference and the appliance of business change strategies generate questions about the selection criteria. This will be the future research axis in order to complete this approach. The appliance of this approach on companies facing changing situations or evaluating their alignment will contribute in the creation of a strategy selection framework. In deed, the selection of adequate strategies depends on the organizational context.

References

1. Wegmann, A.: On the Systemic Enterprise Architecture Methodology (SEAM), Proceedings of the 5th International Conference on Enterprise Information Systems p.483-490 , 2003
2. Paolo, G., Manuel K., Mylopoulos J., Pistore M.: The Tropos Methodology: an overview, Methodologies and Software Engineering for Agent Systems. 2004. Kluwer Academic Publishing
3. Nurcan, S., Barrios, J., Rolland, C.: Une méthode pour la définition de l'impact organisationnel du changement, Numéro special de la revue Ingénierie des systèmes d'Information, 7 :4, 2000
4. Nurcan, S., Rolland, C.: Using EKD-CMM electronic guide book for managing change in organizations, 9th European-Japanese Conference onInformation Modelling and Knowledge Bases, p.105-203, 1999
5. Su, N., Mylopoulos J.: Conceptualizing the co-evolution of organizations and information systems: an agent-oriented perspective, 25th International Conference on Conceptual Modeling - ER, p.296-310, 2006
6. Salinesi, C., Etien, A., Zoukar, I.: Goal / Strategy Maps - Methods, Techniques and Tools to Specify Requirements in Different Evolutionary Contexts, International Conference on Systems Engineering (INCOSE), Toulouse, France, June 2004.
7. Henderson, J., Venkatraman, N.: Strategic Alignment: Leveraging Information Technology for transformig Organization, IBM syst. J.32, 198-221, 1993
8. Regev, G., Wegmann, A.: Remaining Fit: On the Creation and Maintenance of Fit, Workshops (2) p.257-258 , 2004

Towards a Multi-perspective Assessment of Scalability of Distributed IT Services

Zsófia Derzsi and Jaap Gordijn

Free University, FEW/Business Informatics, De Boelelaan 1083a, 1081 HV Amsterdam, The Netherlands. (derzsi, gordijn)@few.vu.nl

Abstract. Scalability is an important issue in distributed IT service design and should be addressed at least from *technical* and *economic* perspectives. We present a conceptual framework that addresses scalability from these two perspectives. To explore scalability of distributed services, our framework employs a value model and UML deployment model to describe scalability concerns of a distributed, commercial IT service. We illustrate our approach by a case study.

1 Introduction

We consider commercial and distributed IT services built on the top of Internet technology as *commercial* deeds of a mostly intangible nature [4]. They are operated by *networked* constellations of enterprises (suppliers and costumers), who, using each other core competencies, jointly work on the satisfaction of an IT-intensive consumer need. The underlying information system architecture, which puts the constellation into operation, shows a distributed landscape as software and hardware components to realize such a service are typically distributed among a number of these enterprises.

The term 'distributed IT service' requires two different perspectives. From the *perspective of business* it is a *commercial* concept, which is offered by a network of enterprises rather than one enterprise to satisfy an IT-intensive consumer need. A well-known example is the need to surf on the web, which can be satisfied by an Internet Service Provider (ISP). From the *perspective of information technology* it means that the required software and hardware components of the information system are located at multiple, different places, which are interconnected e.g by web service technology leveraging the Internet [7]. To avoid confusion, in the rest of this paper we use the term 'service' to label the commercial perspective, while 'web services' refer to the information technology perspective.

Ideally, distributed services should remain both technically and economically feasible in different business settings. Many configurations are possible, caused by e.g. varying the participating enterprises or by the increase of number of consumers. One desirable requirement for the underlying distributed system is that it must be *scalable*, meaning that it should handle different business settings while at the same time provide a constant output in performance [7, 5]. Additionally, business and information technology perspective on a distributed system should match; a business setting should match with its supporting distributed information system with respect to scale.

Assessing the scalability of distributed services is a complex task. Many scientific papers addressed the problem and proposed different analytical and performance measures, mostly from the information system point of view [7, 1]. System scale should be matched by sufficient capacity in soft- and hardware components resulting in financial consequences (e.g. investments) for its stakeholders. Moreover, assumptions for designing a scalable system (e.g. the expected number of customers) are important; a system that should support 5 concurrent customers looks often quite different from a system that must support millions of concurrent customers. Several examples of system scalability research focus on the importance of cost effects [8, 5] in addition to performance. In a commercial, networked business setting, however, the *allocation* of these costs among enterprises is of an importance, too, because such an allocation directly influences potential profitability of an enterprise involved.

In this paper, we propose a conceptual framework to relate scalability concern from a business value perspective (using e^3*-value* models) and information technology perspective (using UML deployment diagrams). Additionally, we discuss how to address scalability analysis from a business *and* technical perspective. We illustrate our framework by a small case study.

2 Perspectives on Evaluating Scalability of Distributed IT Services

Our framework is motivated by the work of Neumann [7], where scalability of distributed information systems is addressed from three aspects: *(a) size*, the increase or decrease of objects or users of the system, *(b) allocation of computational resources*: hard- and software components, as the execution of subtasks can happen by different stakeholders at geographically different locations, and *(c) administrative control*, as systems leveraging the Internet technology can cross multiple, independent administrative domains and conflicting policies can occur with respect to e.g. resource usage or security.

Ideally, the performance of distributed information systems should remain constant as any change occur along these three aspects (e.g. increase of users). Analyzing the effects of these changes purely from the information technology perspective would only judge whether the provision of the distributed service it supports is technically feasible. However, scalability has its price, and thus its financial consequences. It is of an importance to examine the resulting *economic value* effects in order to assess whether the provision of the distributed service remains financially sustainable.

Figure 1 summarizes our framework built on the above articulated concepts and shows their relations. Technical scalability is ultimately about *system performance* and addresses whether it remains constant if there are changes in the three forementioned scalability aspects (users/objects, resource allocation, administrative control). Economic scalability is about *financial performance* of the enterprises of networked constellation, and evaluates the resulting financial effects of these changes (e.g. additional investments). The scale as supported by the distributed information system (technical perspective) should correspond to its financial effects as indicated by the business perspective.

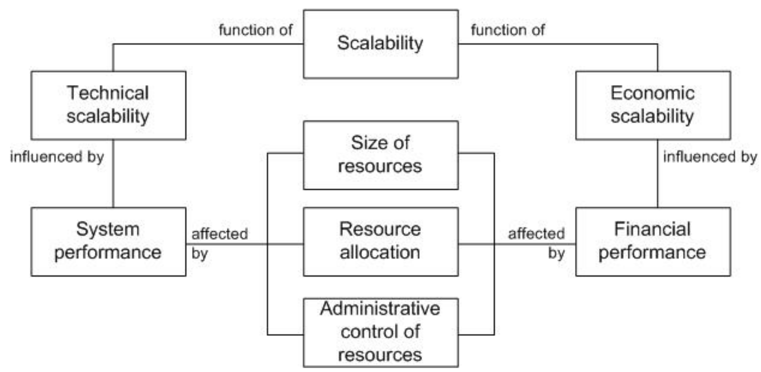

Fig. 1. Conceptual framework to assess scalability of commercial, distributed IT services supported by distributed information systems

To guide scalability analysis, our research employs modeling techniques to describe the information system and the business setting focusing on size, resource allocation, and administrative control. In addition, our research investigates how constructs of these modeling techniques can be related to support reasoning over scalability. As a first step, we employ the e^3*-value* technique (for a detailed description please consult [3]) to describe the business model and we use the UML deployment modeling technique to describe the information system perspective. We explore to what extent these modeling tools are applicable to guide the *technical scalability* assessment, and we also show how constructs of a value model and a deployment model can be related to support our analysis. In this paper, we do not elaborate on economic scalability.

3 Case Study: A Commercial Distributed Service to Reduce Imbalance in Electricity Supply

We now introduce a case-study on electricity supply and consumption (see [2] for details). Due to the physical nature of electricity power, the amount of electricity supplied to the network must be *exactly equal* to the amount of electricity consumed. This balance has to be maintained continuously otherwise power outages will occur. This requirement is ensured by the Transmission System Operator (TSO), who compensates imbalance real-time and charges imbalance fee for the parties, who caused the imbalance.

The analyzed distributed service (Distributed Balancing Service (DBS)) is used to perform near-real time, distributed control over the electricity supply and consumption of commercial portfolios (consisting of a series of electricity generators and consumers) [6] in order to reduce imbalance. In case of imbalance, consumers and/or producers of the commercial portfolio are asked to change their level of production and/or consumption. Obviously, such near real-time control is only possible using advanced, distributed information technology. All stakeholders of the portfolio (producers, consumers and supplier) have to employ certain software and hardware for execution of the DBS at their production and consumption sites.

Portfolios may vary in size (i.e. the number of consumers) and in geographic location (i.e. due to the liberalized electricity market [6]). In most cases, portfolios aggregate stakeholders with a different operational profile (i.e. wind turbines, generators), thus the administrative complexity while executing DBS increases. It is thus important to assess the scalability of DBS to assure its deployment in different portfolio settings. In the following we use this case to exemplify our model-driven analysis, focusing, due to space limitations, on technical scalability.

4 Analysis of Technical Scalability: a Case Study

4.1 Characteristics of the Business Design Affecting Technical Scalability

We have constructed an e^3*-value* business model (we assume that the reader is familiar with e^3*-value* , otherwise please consult [3]), Figure 2 shows a simplified extraction of it as presented in [2]. We now assess how the e^3*-value* method supports the analysis of technical scalability (see Figure 1).

The model represents the *one-time execution* of the functionality offered by DBS, namely keeping the balance of supply and consumption. In the e^3*-value* methodology such an execution is shown by a *dependency path*, which models how a consumer need is satisfied by performing 'value activities' and 'value transfers' by different entities. In our example, the path connects the business parties of the networked constellation (represented by 'actor' and 'market segment' modeling constructs) who jointly execute the distributed service satisfying the occurring business need: the imbalance reduction. 'Value activities' demonstrate *who* executes which activity with respect to the distributed service. 'Value transfers' encapsulate exchanged 'value objects' resulting from performed 'value activities'.

The dependency path of Figure 2 thus depicts that a supplier executes a 'Balancing control' activity to decrease imbalance of 'Supply'. It operates together with the 'Operation control' activity maintained by consumers, which controls their 'Consumption'. Consumers offer their 'Device flexibility' as a result of their 'Operation control' and receive 'Compensation fee' in return.

The path first provides information concerning the *size* of actors of the networked constellation executing the distributed service. An 'Actor' models by definition one (business) entity. Cardinality of the market segment is equal to the number of actors it aggregates. By summing up the number of actors along

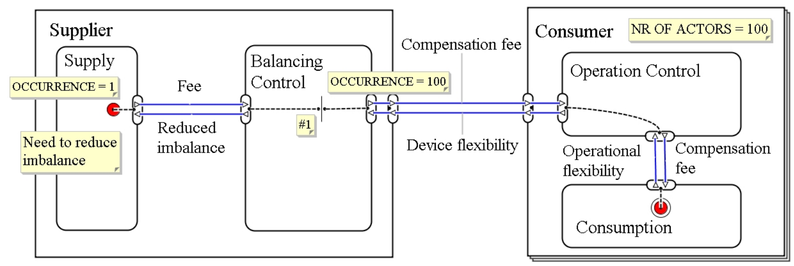

Fig. 2. Structure of the business design, represented by e^3-*value* modeling technique

the path, the number of actors of the constellation can be found for the need at hand (in this example: 100 consumers + 1 supplier = 101). This number of actors is a first indication of scale to be supported by the information system.

The path also helps to determine the number of value transfers between actors, based on the number of occurrences of the need (in this example, 1, since the dependency path demonstrates one-time execution), and influenced by the number of actors. In this example, the number of transfers is sized up according to the number of consumers involved in the service execution, expressed by the explosion element (#1).

In terms of scalability aspects (see Figure 1), the value model helps to articulate the following characteristics of the business setting influencing technical scalability:

- *Size:* the value model shows the number of business actors of the networked constellation and the number of value transfers.
- *Resource allocation:* the value model demonstrates the distributed nature of the service by allocating value activities to different entities, but does not provide any specific information over the employed hard- and software resources, nor how these activities are executed. In addition, it does not show how the business actors of the constellation are geographically distributed.
- *Administrative control:* the value model only shows what value activity is executed and what value object is exchanged but gives no operational insight.

To assess technical scalability of the DBS in more detail, it is necessary to explore the business processes of actors within the networked constellation.

4.2 Characteristics of Information System Design Affecting Technical Scalability

Below, we present a UML deployment diagram to describe the structure of the distributed information system supporting the provision of DBS. The diagram is constructed based on interviews with domain experts. Figure 3 shows an extraction of it (a more detailed model and explanation can be found in [2]). We now assess how the UML deployment diagram supports the analysis of technical scalability (see Figure 1).

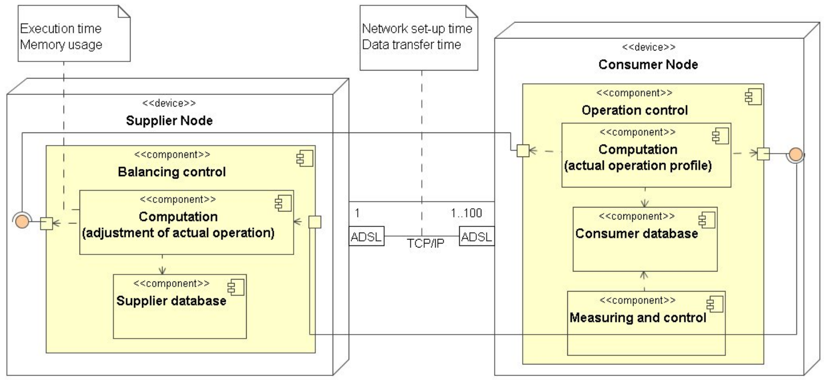

Fig. 3. Structure of the distributed information system, represented by UML deployment class diagram

The depicted deployment diagram gives a better structural insight by showing the *allocated soft- and hardware resources* and *web service ports* offering and requiring web services. The diagram aggregates instances of physical nodes into classes that host the same software and hardware structure, yet independently from geographic location. The cardinality of these classes thus provides information about the *size* of employed hardware and software resources of the information system.

The modeled ports attached to components derived from value activities (see Figure 2) help to understand the invocation of web services of the system. These web services are invoked via a TCP/IP based communication path connecting consumer and supplier nodes. Each node communicates via ADSL router.

The diagram also represents the idea of centralized communication in this specific case. 'Consumer nodes' do not exchange data with each other, they only communicate with the 'Supplier node', highlighting the centralized manner of data sharing. It is the 'Balancing control' component, which possesses all the information needed to adjust actual operation profiles of consumers and thus to perform distributed control. 'Operational control' components are responsible only for the local device control based on the adjustments provided by 'Balancing control'. Such a centralized organization of communication and web service exchanges suggests that the 'Balancing control' component of 'Supplier node' may form a potential performance bottleneck that can limit the technical scalability of the distributed service [9]. We assume that local device control is performed satisfactory.

The structure of the information system thus suggests that the communication network and the computation task of the 'Balancing control' influence the system performance, thus the technical scalability. *Functional parameters* are attached to corresponding modeling constructs (see attached comments in Figure 3). We employ the following metrics: *(a) network set-up time*, the time needed to initialize and to build up automatically the communication network between supplier and consumer nodes, respectively, *(b) data transfer time*, the time needed to transfer data from one point to another, as a function of the type of the end connection (i.e. ADSL), *(c) execution time*, the time needed to perform the required tasks, *(d) memory usage* for computation and for maintenance of communication channels.

We assume that these metrics together determine the *maximum number* of web service invocations that the 'Supplier node' can handle. To determine the value and significance of these performance metrics, however, the analysis of operational processes and of the behavior of web service invocations (e.g. regularity) is essential. To this end, the static modeling approach, as the UML deployment diagram suggests, is not sufficient. In addition, 'network set-up' and 'data transfer time' is location dependent, yet the deployment diagram does not provide insight over the geographic allocation of nodes. Moreover, the analysis of operational processes is essential to assess how the administrative control aspect of scalability would influence system performance, since devices might have diverse operational profiles, yet their operation has to be controlled equally.

In terms of scalability aspects (see Figure 1), the structural constructs of the UML deployment diagram help to articulate the following characteristics of the information system influencing technical scalability:

- *Size:* the cardinality of UML constructs shows the size of employed hard- and software components.
- *Resource allocation:* the deployment class diagram gives better structural insight, yet independently from geographic distribution. Web service ports display the structure of web service exchanges.
- *Administrative control:* the deployment diagram shows the components of the distributed service, but gives no insight to the behavior of these components.

Further refined analysis of the operation of components and of web service invocations is needed to get better insight to technical scalability.

4.3 Relating e^3-*value* and UML deployment diagrams

In the following we show that the e^3-*value* technique and UML deployment class diagram, if correctly related, may contribute to the technical scalability assessment. *Value activities* of the value model that are required for the one-time execution of DBS appear as *components* in the UML deployment diagram. Value activities result in *value transfers* between actors exchanging value objects. This is maintained by offered and received web services between *web service ports* of these components on the information system level.

As a consequence, *value transfers* of the value model encapsulate *web service invocations* needed to execute the distributed service (i.e. the offered 'Device flexibility' is supported by these invocations). The e^3-*value* technique is capable to determine the number of value transfers occurring between actors, which can be used to estimate the number of web service invocations between software components that the underlying information system should handle. As the maximum number of web service invocations that can be handled during the one-time execution of DBS is known for the information system at hand, the number of value transfers may indicate whether technical scalability is violated.

5 Conclusions and future work

In this paper, we proposed a conceptual framework to support the technical scalability assessment of commercial, distributed IT services offered by networked constellations of enterprises. The framework addresses the evaluation of scalability among two - *technical* and *economic* - perspectives taking different aspects (size, resource allocation, administrative control) of scale into account.

This paper focuses on technical scalability. To guide the analysis we employed and related e^3*-value* technique and UML deployment class diagram, and we explored how our model-driven approach can support our aim. As a next step, we extend our analysis by employing behavioral modeling techniques for the scalability assessment.

Another line of research focuses on aspects of economic scalability. Coupling the e^3*-value* and UML techniques seems as a suitable candidate to support our assessment [2], however, further expansion of analysis toward business processes is needed in order to gain better insight on financial consequences of scale.

References

1. Gunnar Brataas and Peter Hughes. Exploring architectural scalability. In *WOSP '04: Proceedings of the 4th international workshop on Software and performance*, pages 125–129, New York, NY, USA, 2004. ACM Press.
2. Zsófia Derzsi, Jaap Gordijn, Koen Kok, Hans Akkermans, and Yao-Hua Tan. Assessing feasibility of it-enabled networked value constellations: A case study in the electricity sector. Conditionally accepted by CAiSE 2007, 2007.
3. J. Gordijn and J.M. Akkermans. Value-based requirements engineering: Exploring innovative e-commerce ideas. *Requirements Engineering Journal*, 8(2):114–134, 2003.
4. Christian Grönroos. *Service management and Marketing: A Customer Relationship Management Approach*. John Wiley & Sons, Chichester, UK., 2000.
5. Prasad Jogalekar and Murray Woodside. Evaluating the scalability of distributed systems. *IEEE Trans. Parallel Distributed Systems*, 11(6):589–603, 2000.
6. Koen Kok, Cor Warmer, and René Kamphuis. The PowerMatcher: Multiagent control of electricity demand and supply. *IEEE Intelligent Systems*, 21(2):89–90, March/April 2006.
7. B.C. Neuman. Scale in distributed systems. In T. Casavant and M. Singhal, editors, *Readings in Distributed Computing Systems*, pages 463–489. 1994.
8. Maarten Van Steen, Stefan Van der Zijden, and Henk J. Sips. Software engineering for scalable distributed applications. In *COMPSAC '98: Proceedings of the 22nd International Computer Software and Applications Conference*, pages 285–293, Washington, DC, USA, 1998. IEEE Computer Society.
9. Maarten van Steen and Gerco Ballintijn. Achieving scalability in hierarchical location services. Technical Report, IR-491, Vrije Universiteit, Department of Mathematics and Computer Science, 2001.

Towards Information Systems Design for Value Webs

Novica Zarvić*, Roel Wieringa and Maya Daneva**

University of Twente, Department of Computer Science, Information Systems Group
P.O. Box 217, 7500 AE Enschede, The Netherlands
{n.zarvic,r.j.wieringa,m.daneva}@ewi.utwente.nl

Abstract. In this paper we discuss the alignment between a business model of a value web and the information systems of the participating companies needed to implement the business model. Traditional business-IT alignment approaches focus on one single company, but in a value web we are dealing with various independent businesses. Since a value web is actually a web of services, delivered by IT systems owned by different companies, to ensure alignment we need to specify the services and their properties and then map them on the available IT support in the different companies. Such mappings have to be evaluated in terms of their impact on the profitability of participating in the value web of the different companies. We propose techniques to map services to IT support and show how to do commercial trade-offs.

1 Introduction

In the past decade, the problem of business-IT alignment has become considerably more complex than it was before, because businesses now cooperate in value webs in which they must align their IT services to each other. By values webs we mean networks in which profit-and-loss responsible businesses provide e-services to each other or to consumers [1]. Traditional approaches to business-IT alignment, such as information systems (IS) planning methodologies [2,3] were designed for single companies, but not for value webs [4]. In our research we represent business models graphically by using the e^3-*value* methodology [5,6]. An e^3-*value* model is, thus, representing a value web that we will use to represent e-services. These e-services need to be analyzed and then mapped to functional and quality specifications of relevant information systems. Respectively, any change in the specification of these systems, e.g. because some implementations may be too expensive, may lead to an adjustment of the value model. Figure 1 gives an overview on the models involved in our research. We study the following design questions: How are e-services represented in e^3-*value* and mapped to IS functions? How to relate e-service quality properties to IS quality properties? In which situations to adjust the value model (or even drop its implementation)?

In Sec. 2 we describe the relationship between services and IS properties. In Sec. 3 we show how to use e^3-*value* to represent value webs by means of an working example.

* supported by the Netherlands Organisation for Scientific Research (NWO), project 638.003.407 (Value-based Business-IT Alignment)

** supported by the Netherlands Organisation for Scientific Research (NWO), project 632.000.000.05N01 (Collaborative Alignment of cRoss-organizational ERP Systems)

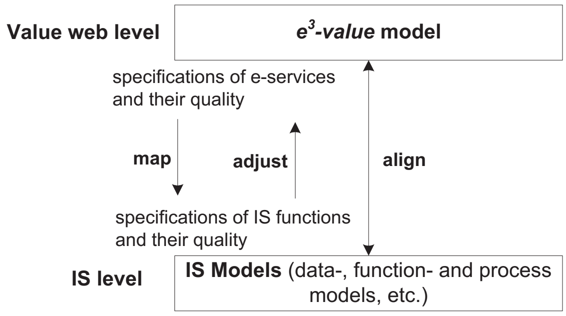

Fig. 1. Aligning value webs with IS models.

Section 4 discusses how to map e-services to IS functions, characterizes the relationship between service quality and software quality, and further describes possible impacts on the value web. In Sec. 5 we conclude the paper.

2 Services and IS properties

A service is defined to be an interaction between a service provider and a service client that has value for the client (who usually offers something of value, such as money, in return) [7,8]. Examples of services are cleaning services, a haircut or the provision of a taxi ride. We define an e-service to be a service delivered over an electronic network. An example is an internet radio service [9], where multiple businesses act together in order to satisfy a customer need. Web Services are often considered to be e-services [10], but they are implementation mechanisms of machine-to-machine interactions over a network. E-services are implemented by means of software systems. We define the *function* of a software system as the interaction between the software and its environment, triggered by some event and with an added value for some stakeholder in the environment. Examples are answering a query (trigger is a question by a user, the answer has added value to the same user), reordering an item when the stock is too low (trigger is a condition change, the reorder is of value for the shop owning the software product that does the reordering), or producing a periodic report (trigger is a tick of the clock, report is presumably useful for whomever reads it). We call all properties of a software system that are not functions and yet have an added value for some stakeholder *quality attrributes* (often called non-functional attributes). Typically, quality attributes are properties of functions. For instance, answering a query should follow a certain response and processing time (time behaviour), reordering an item should prevent unintended access and resist deliberate attacks (security), or producing a periodic report should happen according to conventions or regulations in law (compliance). *Time behaviour*, *security* and *compliance* are some of the quality attributes specified in the ISO 9126 standard [11]. In addition to software quality properties (at IS level) there are service quality properties (at value web level). We define *service quality* as being any service property that adds value to the service. The quality of a service is whatever the client perceives it to be [12]. For instance, a potential reader will not be interested in an online-article, if the download time would take hours. Service quality properties need to be realized by software quality properties. In Sec. 4 we show how they relate to each other.

3 Using e^3-*value* to represent value models

We will illustrate the challenge of mapping e-services to IT support, and of adjusting e-services to available IT support, by means of the small example shown in Fig. 2.

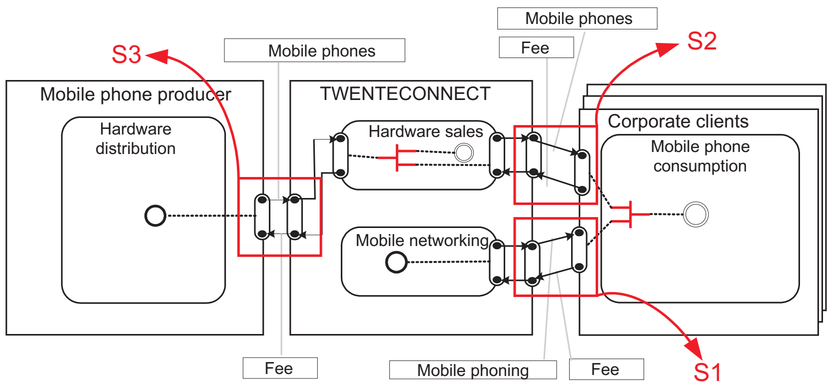

Fig. 2. The TwenteConnect case (regional mobile phoning).

Consider a small telephone company named TwenteConnect, that serves a regional market. The company has been providing so far only fixed land-line services and did not sell any hardware components such as cell phones to their customers. Now, TwenteConnect wants to expand to the area of mobile phone services, again in the same region. Their expansion plan says that before starting to target private clientele, they will run a test phase with corporate clients. The goal is to provide the local police and the staff of the local hospital with mobile phone connections, including mobile phones. As far as TwenteConnect does not produce the hardware (mobile phones) on its own, it relies on a collaboration with a mobile phone producer. In e^3-*value* a value model[1] shows "*who* exchanges *what* with *whom* and expects *what* in return" [13]. It focuses on the business actors and on the reciprocal transfer of value objects between the actors. Note that in Fig. 2 we have non-physical value objects like the mobile phoning provision, but also physical value objects like the mobile phones. This is not uncommon for a 'web of services' and many such constellations consist out of a combination of goods and services [14, p.140]. Following our service definition from Sec. 2, the visible provider/client interactions in e^3-*value* are the value object transfers. So, each value object transfer is a candidate for an e-service.

[1] See http://www.e3value.com/

4 Aligning e-services to IS properties

4.1 Mapping e-Services to IS properties

How are e-services represented in an e^3-value *model?* We call the service delivered to the consumer the *consumer service* of the value web. A scenario path shows how the consumer service is decomposed into services delivered by actors in the value web to each other. In the simple example of Fig. 2 we can identify six value transfers. As far as the *e^3-value* methodology is based on the principle of *economic reciprocity*, we need to consider this rule. This means that a service is represented by at least a value transfer from provider to client, and a value transfer back from client to provider. Such a combination is describing the reciprocal provider/client interaction and is usually grouped into one value interface at each actor. This way of reasoning allows us to separate between provider-specific activities and client-specific activities to be performed. By following the scenario path starting from the corporate clients we can identify three services:

– *mobile phoning*, labeled as S1 in Fig. 2
– *mobile phone delivery*, labeled as S2 in Fig. 2, and
– *mobile phone ordering*, labeled as S3 in Fig. 2.

Consider as an example e-service S1 that consists of two value object transfers which need to be supported by the IS. Now, we can further decompose the activities of S1 into (a) the *provider-specific activities* of the value transfer from TwenteConnect to the corporate clients and (b) into the *client-specific activities* of the value transfer from the corporate clients to TwenteConnect. To identify the IS functions/activities needed to realize the services S1, S2 and S3, we propose to use Porter's value chain as a reference point [15]. It is a generic enough description of activities in any business, and thus, allows us to derive the first list of IS functions/activities needed in support of the services. For instance, provider-specific activities are mainly to be found in the sets of operation and outbound activities in Porter's reference model. Client-specific activities are mainly placed in the set of inbound activities. For further refining these activities we may use for instance function refinement trees [16] or even state diagrams "for finding missing or obscure functions" like suggested by Lauesen [17]. Note that note all identified services in our example are pure e-services. For instance, S3 implies next to the activities taking place over the electronic network also the delivery of mobile phones from the mobile phone producer to TwenteConnect. Note further that S1 and S2 are two independent services offered and provided to the corporate clients. In case we would bundle S1 and S2, we would have to consider the service bundle as one service instead of two.

How to relate service quality attributes to software quality attributes? Clearly, the services in a value web, are realized by the IT systems of the actors. The attainment of service quality in services became an active research area in service marketing [18], which resulted in the definition of different service quality models [12]. The quality of a service in the value web depends on the quality of its enabling IT systems, which explains the relationship between service quality and software quality. In our research we considered at IS level the ISO 9126 standard on software quality [11]. We assume that the reader is familiar with ISO 9126. At service level we considered (i) early results of work by Parasuraman et al. [18], which resulted later in the so-called SERVQUAL model [19], and (ii) recent work by O'Sullivan et al. on service properties [20].

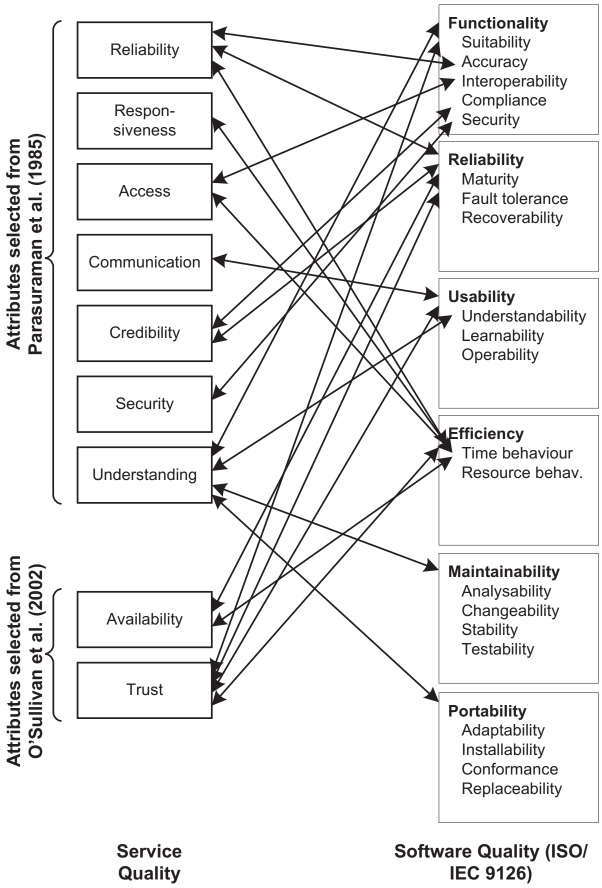

Fig. 3. Relating Service Quality to Software Quality.

This led to the selection of service quality attributes presented in Fig. 3 on the left. We used our service quality definition as a criterion for the selection. We noticed that O'Sullivan discusses many properties, e.g. obligation for payment, penalties, etc. These, however are rather business rules and do not add value to a service. Furthermore, the relationships between service quality attributes and software quality attributes are based on the attribute definitions of the quality models. We analyzed each definition in [18] and [20] and mapped it to those elements of [11], which had same or very similar meaning. The arrows in Fig. 3 represent those mappings. Note that a box on the left can be mapped either to a box on the right (representing a set of software quality attributes), or directly to a single software quality attribute inside a box. In the following we describe shortly the service quality attributes and how they relate to ISO software quality attributes. *Reliability* involves consistency of performance and dependability and some of these aspects can be found at software level in the set of reliability attributes, but also at accuracy and time behaviour. *Responsiveness* concerns the willingness or readiness to provide services. It involves timeliness of service. *Access* involves ease of contact and can be mapped to interoperability and time behaviour. *Communication* means keeping customers informed in language they can understand and can directly be mapped to understandability in ISO 9126. *Credibility* involves trustworthiness, believability and honesty and relates to certain aspects of compliance, but also to the set of reliability.

Security can directly be mapped to the software security quality attribute. *Understanding* involves making the efforts to understand the client's needs and analogies can be found in several sets of ISO 9126. The *availability* of a service is the times when and places where the service is available. This affects time behaviour and several reliability issues. *Trust* is an attribute that deals with trusting the competence and intentions of a service provider. It does not have a direct counterpart in ISO 9126, but relates to several software attributes as can be seen from the figure.

For instance, TwenteConnect might have implemented a direct debiting system for collecting the fees for service S1. To add value to the functionality TwenteConnect decided to collect the fees via a secure communication channel. So the quality attribute *security* for the payment of the fees adds value to service S1. This property can directly be related to *security* at software level and should be considered in the IS design process. Now, consider for instance *fault tolerance* from the set of *reliability* properties. Fault tolerance are those properties that influence the ability to maintain a specified level of performance in cases of software faults. The point is that we cannot have a high performance (e.g. quick responses) if there are too many security checks, each of which takes its time to be completed. In no way security should be comprised to performance. This example makes clear that quality properties should not be considered solely and that it should always be determined which impact they have among each other.

Note that Fig. 3 is actually a set of hypotheses: Each bidirectional arrow is a hypothesis and states the service attribute has impact on the IS attribute. So, each arrow should be elaborated to give guidelines about what exactly this impact might be. Note that there might also be relationships between service attributes and software attributes that are not assigned in the figure, because the similarities are based solely on the attribute definitions. In future work we plan to research these relationships.

4.2 Adjusting Value Models to IS properties

IS design for value webs implies the identification of IS functions. It is desired to reuse available systems. It may turn out that there does not exist available systems capable for realizing functions, which in turn results in the need to design or buy new systems. Such an investment needs to be evaluated financially. Currently two techniques are supported by the e^3*-value* tool for assessing economic sustainability of a value web, namely *net value flow* and *discounted net present cash flow* technique (DNPC), which is based on the well-know net present value (NPV) technique. In the context of our working example, we want to evaluate whether the test phase promisses a positive net value flow for TwenteConnect. We consider that the local hospital needs 20 and the police needs 80 mobile phone connections and mobile phones, so in total 100. For each mobile phone TwenteConnect has to pay 40 Euros to the phone producer (100*40€=4.000€), but sells it for 1 Euro to its corporate clients (100*1€=100€). TwenteConnect sells connectivity as a monthly flatrate for 15 Euros (100*15€=1.500€/month). If we consider the time-period of one year we can assume to get a net income of 14.100€(-4.000€+18.100€=14.100€). Note that the second year will differ in such a way that the income will be 18.000€, because the corporate clients already have mobile phones and we assume two years of average usage of such hardware. So far we did not address the time value of money, but for doing so we can use the DNPC. Take the first time-period were we calculated an undiscounted net value flow of 14.100€. By discounting it, let's say with an interest rate of 5%, we have a value at the start of the first period of just 13.428,57€. If we discount the net value flow for the second year ($18.000/1.05^2$), the value will at the start of the first time-period be just 16.326.53€. The DNPC approach allows to include expenses for *investments*. We might find out that the functionality of available systems does not suffice to realize participation in the value web. In our case we would need to make an investment for a software piece amounting to 3525€, for realizing the business case. In terms of the DNPC this is called an upfront investment, where a special time-period 0 has to be introduced.

Period	Revenues	Expenses	Investments	Net value flow	DNPC
0			3.525	-3.525	-3.525
1	18.100	4.000		14.100	13.428,57
2	18.000			18.000	16.326,53
Total				**28.575**	**26.230,10**

Table 1. Comparing evaluation approaches: net value flow vs. DNPC

Table 1 compares the (undiscounted) net value flow calculations with the DNPC for the two mentioned years (period 1 and 2) with an upfront investment period 0 to include the investment. We recommend the usage of DNPC in order to get a more realistic picture of the economic situation, because it discounts future profit. The investment in this case was so small that it had no impact on the business constellation, but there are other examples conceivable, where the investment exceeds the profit. Now, suppose again that the functionality of available systems does not suffice for realizing the business idea of Fig. 2. TwenteConnect would have following possibilities: (i) developing or buying a new system, (ii) adjusting the value model, and (iii) revoking participation. In case TwenteConnect would need an investment of 35.000€, it would almost three years run negative numbers. As a result TwenteConnect would probably not commit the huge investment of buying or developing a new system. As a result, TwenteConnect would need to analyze whether the value model can be adjusted to supported IS functions of its available systems, and how this would differ from the initial e^3*-value* model. The worst case would appear, if (i) and (ii) are both not feasible. Then possibility (iii) would step in, which means that TwenteConnect would drop the idea of participating in the value web.

5 Conclusions and Future Work

In this paper we addressed how to perform IS design for value webs, by identifying services and functions from the e^3*-value* model and mapping them to the IS level. We also considered quality attributes at both levels and indicated how these could be related. In future research we will investigate some relations in Fig. 3 in more detail, to research the impact, and what guidelines we can derive from that. We will investigate these issues by means of performing case study-oriented research with our industrial partners.

References

1. R. Wieringa, J. Gordijn, and P. van Eck: Value-based Business IT Alignment in Networked Constellations of Enterprises. In: Proceedings of the 1st Internation Workshop on Requirements Engineering for Business Needs and IT Alignment (REBNITA 2005). (2005)
2. J. Martin: Strategic Data Planning Methodologies. Prentice Hall, New Jersey (1982)
3. C. Finkelstein: An Introduction to Information Engineering - From Strategic Planning to Information Systems. Addison Wesley, Sydney (1989)
4. N. Zarvić and M. Daneva: Challenges and Solutions in Planning Information Systems for Networked Value Constellations. In M. Weske and M. Nüttgens, ed.: Proceedings of the EMISA 2006 workshop. Volume P-95 of LNI - Lecture Notes in Informatics., Hamburg, GI - Gesellschaft für Informatik (2006) 119–131
5. J. Gordijn and H. Akkermans: Designing and Evaluating E-Business Models. IEEE Intelligent Systems **16**(4) (2001) 11–17
6. J. Gordijn and H. Akkermans: Value-based requirements engineering: exploring innovative e-commerce ideas. Requirements Engineering Journal **8**(2) (2003) 114–134
7. W. J. Regan: The Service Revolution. Journal of Marketing **27**(3) (1963) 57–62
8. IBM: Services Sciences, Management and Engineering. `http://www.research.ibm.com/ssme/services.shtml` (2004)
9. J. Gordijn, E. Yu, and B. van der Raadt: e-Service Design Using i* and e3value Modeling. IEEE Software **23**(3) (2006) 26–33
10. T. Pilioura and A. Tsalgatidou: E-Services: Current Technology and Open Issues. In: Proceedings of the Second International Workshop Technologies for E-Services (TES 2001), Springer (2001) 1–15
11. ISO 9126-1: Information Technology - Software quality characteristics and metrics (1995)
12. C. Grönross: Service Management and Marketing - A Customer Relationship Management Approach. 2nd edn. John Wiley, Chichester (2000)
13. J. Gordijn: Value-based Requirements Engineering: Exploring innovative e-Commerce ideas. PhD thesis, Free University of Amsterdam (2002)
14. N. Slack, S. Chambers, C. Harland, A. Harrison, and R. Johnston: Operations Management. Pitman Publishing (1998)
15. M. E. Porter: Strategy and the Internet. In: Harvard Business Review. (2001)
16. R. Wieringa: Design Methods for Reactive Systems. Morgan Kaufmann, San Francisco (2003)
17. S. Lauesen: Software Requirements: Styles and Techniques. Pearson Education Limited, Harlow, England (2002)
18. A. Parasuraman, V. A. Zeithaml, and L. L. Berry: A Conceptual Model of Service Quality and Its Implications for Future Research. Journal of Marketing **49** (1985) 41–50
19. V. Zeithaml, A. Parasuraman, and L. Berry: Delivering Quality Service - Balancing Customer Perceptions and Expectations. The Free Press, New York (1990)
20. J. O'Sullivan, D. Edmond, and A. ter Hofstede: What's in a Service? Towards Accurate Descriptions of Non-Functional Service Properties. Distributed and Parallel Databases **12** (2002) 117–133

Collaborative IT policy making as a means of achieving Business-IT alignment

J. (Josephine) Nabukenya, P. (Patrick) van Bommel, H.A. (Erik) Proper

Institute for Computing and Information Sciences, Radboud University Nijmegen
Toernooiveld 1, 6525 ED Nijmegen, The Netherlands
{J.Nabukenya, P.vanBommel, E.Proper}@cs.ru.nl

Abstract. This paper is concerned with the application of *collaboration engineering* to improve the quality of *policy-making processes* as they occur in a *business-IT alignment* context. Policies are needed to guide complex decision-making. The creation of such policies is a collaborative process. The quality of this collaboration has a profound impact on the quality of the resulting policies and the acceptance by its stakeholders. We therefore focus on the use of techniques and methods from the field of collaboration engineering to improve the quality of Business-IT alignment related policy-making processes.

1 Introduction

Alignment of business and IT starts with the alignment of their respective underlying policies [2, 3]. The alignment of these policies entails a collaborative effort involving representatives from both IT and business domains. In this paper, we are concerned with collaborative policy making processes as a means to achieve business and IT alignment by starting at the policy level. In general, a policy [4] is a guide that establishes parameters for making decisions; it provides guidelines to channel a manager's thinking in a specific direction.

Policies are created in a *policy-making process*, which involves an iterative and collaborative process requiring an interaction amongst three broad streams of activities: problem definition, solution proposals and a consensus based selection of the line of action to take. The core participants of a policy-making process must be involved in complex and key decision making processes themselves, if they are to be effective in representing organizational interests. In the case of business-IT alignment, key decision makers from at least both IT and business side (but potentially also human-resources, finance, etc) should be involved. Obtaining specific, well understood, and committed to, policies are a key indicator for successful organizational decision-making.

In essence, a policy-making process is a collaborative design process whose attention is devoted to the structure of the policy, to the context and constraints (concerns) of the policy and its creation process, and the actual decisions and events that occur [5]. We aim to examine, and address, those concerns that have a collaborative nature and are related to Business-IT alignment issues. Such concerns include the involvement of a variety of actors resulting in a situation where multiple backgrounds, incompatible interests, and diverging areas of interest all have to be brought together to produce an acceptable policy result. These collaborative challenges come particularly to the fore in the case of business-IT alignment.

2 Collaborative policy making processes

The concept of *policy* has been defined by several researchers [6, 7, 8, 10]. It is beyond the scope of this paper to provide a full survey of these definitions. However, based on the definitions of these researchers, we use the following integrated definition of a policy: *a policy is a purposive course of action followed by a set of actor(s) to guide and determine present and future decisions, with an aim of realizing goals*. In a Business-IT alignment context, the policies of the Business and IT domains will have to be aligned.

According to [5], the process of *policy-making* includes the manner in which problems get conceptualized and are brought to a governing body in order to be resolved. The governing body then formulates alternatives and select policy solutions; and those solutions get implemented, evaluated, and revised.

In shaping the collaborative nature of policy making processes, we turn to the field of *collaboration engineering*. Essentially, this field revolves around the use of *information and communication technologies* to enable the collaboration between people. Although organizations have tried to collaborate in their organizational processes to achieve maximum value from their efforts, achieving effective team collaboration still remains a challenge. Collaboration is the degree to which people in an organization can combine their mental efforts so as to achieve common goals [11]. What is needed is the design of effective collaboration processes. This can be achieved by following the collaboration engineering approach which is defined [12] as "the design of re-usable collaboration processes and technologies meant to engender predictable success among practitioners of recurring mission-critical collaborative tasks".

The choice for developing a collaborative IT policy-making process to achieve Business-IT alignment using a collaboration engineering approach is based on a number of reasons. The major reason for us to take this approach, is that creating policies is a searching and iterative problem-solving collaborative work; this may require external support from professional policy developers / facilitators. These are commonly found to be expensive and scarce. CE seeks to bring the value of facilitated interventions to people who do not have access to facilitation.

Collaboration engineering researchers have identified five general patterns of collaboration to enable a group to complete a particular group activity [12]: i) *Diverge* – to move from a state of having fewer concepts to a state of having more concepts. The goal of divergence is for a group to create concepts that have not yet been considered; ii) *Converge* – to move from a state of having many concepts to a state of having a focus on, and understanding of, fewer concepts worthy of further attention. The goal of convergence is for a group to reduce their cognitive load by reducing the number of concepts they must address; iii) *Organize* – to move from less to more understanding of the relationships among the concepts. The goal of organization is to reduce the effort of a follow-on activity; iv) *Evaluate* – to move from less to more understanding of the benefit of concepts toward attaining a goal relative to one or more criteria. The goal of evaluation is to focus a discussion or inform a group's choice based on a judgment of the worth of a set of concepts with respect to a set of task-relevant criteria; v) *Build Consensus* – to move from having less to having more agreement among stakeholders on courses of action. The goal of consensus building is to let a group of mission-critical stakeholders arrive at mutually acceptable commitments.

The patterns of collaboration do not explicitly detail how a group could conduct a recurring collaboration process, especially with teams who do not have professional facilitators at their disposal. This can be achieved by the key collaboration engineering concept: *the thinkLet*. A thinklet is defined by [12] as "*the smallest unit of intellectual capital required to create a single repeatable, predictable pattern of collaboration among people working toward a goal*". ThinkLets can be used as conceptual building blocks in the design of collaboration processes. Some examples of thinkLets are provided in table below. More examples of thinkLets can e.g. be found in [13].

ThinkLet Name	**Pattern**	**Purpose**
DirectedBrainstorm	Generate	To generate, in parallel, a broad, diverse set of highly creative ideas in response to prompts from a moderator a moderator and the ideas contributed by team mates.
BucketSummary	Reduce & clarify	Remove redundancy and ambiguity from generated items.
BucketWalk	Evaluate	To review the contents of each bucket (category) to ensure that all items are appropriately placed and understood.
MoodRing	Build Consensus	To continuously track the level of consensus within the group with regard to the issue currently under discussion.

3 Design and evaluation of policy-making process

In this section, we present how our research was conducted and evaluated. We will do so in terms of a description of the research approach and cases involved. We also present a description of the generic collaborative Business-IT policy-making process, and relate this to the results of the case studies.

The aim of our research was to establish how to realize a "good Business-IT policy" in a collaborative process and how this process can be improved by support of collaboration engineering in order to achieve Business-Business-IT alignment. To develop and evaluate our collaborative policy-making process, we followed the *action research* methodology process proposed by [14] where four activities that can be carried out over several iterations (in our case two) are involved. The 'Plan' activity is concerned with the exploration of the research site and the preparation of the intervention. The 'Act' activity involves actual interventions made by the researcher. The 'Observe' activity is where the collection of data, enabling evaluation, is done during and after the actual intervention. Finally, the 'Reflect' activity involves analysis of collected data and infers conclusions regarding the intervention that may feed into the 'Plan' activity of a new iteration.

We used action research because it is an applied research method that can be tested in the field. Better still, it addresses the “how to” research questions as seen in our research aim. More so, the continuous design and evaluation of collaborative processes may not be easy to study in a constructed setting. Lastly, action research allowed us to evaluate and improve our problem-solving techniques or theories during a series of interventions.

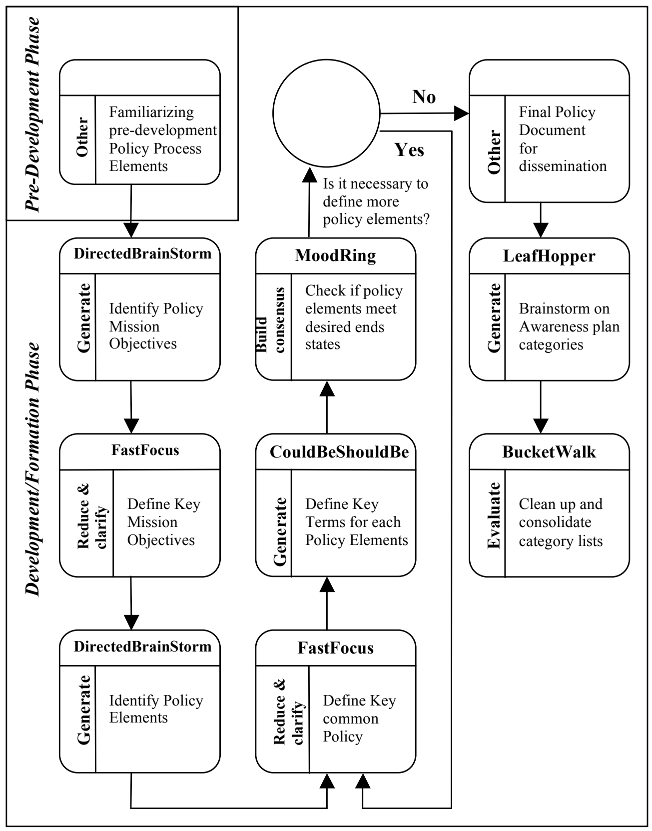

Fig. 1. Collaborative IT Policy-Making Process Design

Two Business-IT policy development workshops using the collaboration process were run. The experiences from each workshop resulted in changes to the design of the final collaboration process. In the *first case*, a team of five experienced Business-IT workers and involved in making policies for the Business-IT Department of the Ministry of Finance, Planning and Economic Development (MOFPED), Uganda used the process to develop an Business-IT policy. The *second case* involved a team of sixteen people comprised of two experienced Business-IT workers involved in Business-IT policy-making and fourteen Master’s Students (2nd year, Computer Science) at Radboud University Nijmegen

(RUN), the Netherlands, used the process to develop an Business-IT policy in form of architectural principles for the student portal information system for RUN.

To evaluate the performance and perception of the collaborative process by the participants, we collected and analyzed explorative data using three kinds of instruments: observations, interviews and questionnaires comprising of qualitative and quantitative questions. In particular, we investigated the effectiveness; efficiency; and policy stakeholders' satisfaction with the collaborative Business-IT policy process and its outcomes; policy elements identification; the degree of applicability of the Business-IT policy process. The need to realize a quality IT policy from a collaborative effort to achieve Business-IT alignment is the basis for the design of the collaborative IT policy-making process (Figure 1). The collaborative process was designed following the collaboration engineering approach described in Section 2. Even though this approach comprises several design steps, the ones relevant to our research study included decomposing the process into collaborative activities, the classification of these activities into patterns of collaboration, selection of appropriate thinkLets to guide facilitation of the group during the execution of each activity as well as making the design process more predictable and repeatable. Below we give a description of the criteria we followed to evaluate the performance of the process, and a presentation of the final design of the process, respectively.

The design of the collaborative process was derived from two iterations based on a selected design criteria. The criteria selection was derived from the goal of the collaboration process. The collaboration process goal aimed at addressing how to realize a quality Business-IT policy using a repeatable collaborative process. The following four criteria were considered by us: (i) *effectiveness* – the collaborative Business-IT policy-making process should enable Business-IT policy-making stakeholders to achieve their goal, (ii) *efficiency* – the collaborative Business-IT policy-making process should take stakeholders less time for attainment of the Business-IT policy than without the use of a collaborative approach, (iii) *degree of applicability* – the extent to which the collaborative policy process can be applied to varying Business-IT policy types and (iv) *perceived policy elements identification* the collaborative Business-IT policy-making process should enable stakeholders to have a common understanding of the Business-IT policy elements (and their definitions).

The collaborative policy process underwent two iterations prior to deriving the final process design. The two iterations of the earlier versions of the process were applied in the two cases described above. The final process design is shown in Figure 1 in which we present the steps required to develop/form an IT policy document, the patterns of collaboration with related thinkLets used to guide the group to execute each step. The identification and choice of thinkLets to enable us evaluate and achieve the process goal can be seen in [13].

The process is divided into two main phases: a *pre-development phase* and a *development phase*. The first phase starts with the participants familiarizing themselves with and agreeing on the pre-development elements gathered in several earlier pre-meetings. Actual development of the policy is based on these elements. The elements comprise the problem to be solved; the relevant information to be used to develop the policy; a legal framework to support the policy to be developed; the ownership of the policy; leadership positioning i.e. who is to spearhead the process; who are the stakeholders (internal and external); technical resources to facilitate the process.

The next brainstorm activity, guided by the DirectedBrainstorm thinkLet, involves participants identifying relevant policy objectives. The result from this activity is a brainstormed list of Policy Mission Objectives. In the ensuing activity, using the FastFocus thinkLet, participants organize the brainstormed list by extracting only the key policy Mission Objectives. They do this by grouping ideas and eliminating any redundancies. The result from this activity is a cleaned list of Key Policy Mission Objectives.

In the activity that follows, guided by the DirectedBrainstorm thinkLet participants are asked to identify and agree on common policy elements definitions that suit the Key Mission Objectives. The result of this activity is a brainstormed list of policy elements. Using the FastFocus thinkLet, the participants organize (clean-up) the resulting brainstormed list by extracting only the common elements. The result of this activity is a cleaned list of Key Policy Elements.

The activity that follows involves defining the Key terms for each of the policy elements defined. Using the CouldBeShouldBe thinkLet, participants brainstorm terms that they 'could' consider as appropriate for each policy element. Later, participants are then propose a term that they 'should' take as Key to each policy element.

The activities above result into a Policy document. Using the MoodRing thinkLet, participants are required to reach consensus. They do this by voting on a YES/NO basis, where a YES is voted if the elements definitions and terms meet the desired end states and a NO if it does not. A verbal discussion is held until some sort of consensus on the final policy document is reached.

Finally, the policy stakeholders need to plan how they will communicate the policy document to its intended users/owners. In this activity, they are required to draw up a policy awareness plan. Two ways are pre-determined that can be used, i.e. communication and education. Using the LeafHopper thinkLet, participants brainstorm about ways in which each of these can be addressed. The brainstormed lists are evaluated to remove any redundancies. This is achieved by using the BucketWalk thinkLet.

The evaluation of the collaborative policy-making process design was implemented following a manual procedure. We used the Microsoft Word (MSWord) tool, an LCD projector, removable disks and voting sheets (paper-based) to implement the process. Results from the cases are presented in the section below.

To measure the *efficiency* construct, we considered the execution duration of each stage of the process; also how well the participants understood the process tasks for successful execution; and on the whole also considered the time it took the participants to come up with the final policy document and the awareness plan.

Based on our observations, we concluded that the policy process execution time was efficient. It took about an hour and fifteen minutes to execute the process in each of the workshops. That is, the participants managed to execute the process within the duration that was assigned to each stage. Even though the majority of the participants felt that the process execution was efficient, not all were happy with this time length; some required that more time should have been assigned to particular activities such as policy elements identification.

We measured the *policy formation effectiveness* construct by how well the participants managed to come up with a policy at the end of the policy process execution. From our observations, it was noted that the participants effectively managed to form policies with respective awareness plans. This was demonstrated during the consensus stage of the process. Based on the feedback from the voting sheets, it was observed that the participants achieved fairly satisfactory results. This produced the following results:

	Yes	No
Case 1	4 (80%)	1 (20%)
Case 2	12 (75%)	4 (25%)

Having arrived at a complete policy document during the consensus stage, the participants also perceived it as having a common understanding of the policy elements identification.

To measure the *degree of applicability of Business-IT policy process*, we applied the policy process to two cases with different policy types. These included formation of an Business-IT policy, and Architectural Principles for an Information System. It was observed that the policy process was flexible in terms of its applicability in formation of two different types of policies.

To measure the *policy stakeholders' satisfaction* construct, we used the 7-point Likert scale general meeting survey questionnaire where participants can strongly disagree to strongly agree. The instrument validation and theoretical underpinnings can be seen in [15]. The results provided below, indicate that the participants were reasonably satisfied with the policy process outcomes, and the process by which the policies were formed.

	1	2		1	2
Satisfaction with process			**Satisfaction with outcome**		
Score	4.800	3.838	Score	5.160	4.363
Standard deviation	1.376	0.995	Standard deviation	1.310	1.094

4 Conclusions and further research

This paper focussed on the the application of *collaboration engineering* to improve the quality of business-IT alignment related *policy-making processes*. We presented the results of two case studies conducted, regarding the use of collaboration engineering in the context of a policy making processes for business-IT alignment purposes. Based on the results, the quality of the generic policy making proces, in terms of its effectiveness, efficiency and applicability, proved to be a success. As such, the collaborative process has indeed the potential to support organizations in developing quality policies.

As a next step, we aim to more explicitly rationalize design decisions taken in policy making processes. We aim to do so by explicitly relating the goals of the policy making process (its *why*, such as improved Business-IT alignment), the requirements on the process following from these goals (its *what*), the situation in which it needs to be executed (its *within*), to the construction of the policy making process (its *how*). In doing so, we will draw on past results concerning modeling processes [16, 17] since a policy making process can essentially be regarded as a collaborative modeling process.

References

1. Henderson, J., Venkatraman, N.: Strategic alignment: Leveraging information technology for transforming organizations. IBM Systems Journal **32** (1993) 4–16.
2. Keen, P.: Shaping the Future – Business Design Through Information Technology. Harvard Business School Press, Boston, Massachusetts (1991).
3. Tapscott, D., Caston, A.: Paradigm Shift – The New Promise of Information Technology. McGraw–Hill, New York (1993).
4. Robbins, S., Bergman, R., Stagg, I.: Management. Prentice Hall Australia Pty Ltd., Prentice–Hall, Sydney (1997).
5. Sabatier, e.: Theories of the Policy Process. West view Press, Boulder, Co. (1999).
6. Rose, R.e.: Policy Making in Great Britain. Macmillan, London, UK (1969).
7. Friedrich, C.: Man and His Government. Wiley, New York (1963).
8. Anderson, J.: Public Policy–making. Praeger, New York (1975).
9. Eulau, H., Prewitt, K.: Labyrinths of Democracy. Bobbs–Merrill, Indianapolis (1973).
10. Schneider, A., Ingram, H.: Policy Design for Democracy. University Press of Kansas, Lawrence, Kansas (1997).
11. Nunamaker, J., Briggs, R., Vreede, G.d.: From Information Technology To Value Creation Technology. In Dickson, G., DeSanctis, G., eds.: Information Technology and the Future Enterprise, Piscataway, New Jersey, IEEE Comp. Soc. Press (2001).
12. Briggs, R., Vreede, G.d., Nunamaker, J.: Collaboration Engineering with Thinklets to Pursue Sustained Success with Group Support Systems. Journal of MIS **19** (2003) 31–63.
13. Vreede, G.d., Fruhling, A., Chakrapani, A.: A Repeatable Collaboration Process for Usability Testing. In Dickson, G., DeSanctis, G., eds.: Proceedings of the 38th HICSS, Piscataway, New Jersey, IEEE Computer Society Press (2005).
14. Zuber-Skerritt, O.: Action research for change and development. Gower Publishing, Aldershot (1991).
15. Briggs, R., Reinig, B., Vreede, G.d.: Meeting Satisfaction for Technology Supported Groups: An Empirical Validation of a Goal-Attainment Model, Small Group Research, in Press (2006).
16. Hoppenbrouwers, S., Proper, H., Weide, T.v.d.: A Fundamental View on the Process of Conceptual Modeling. In: Conceptual Modeling – ER 2005 – 24 International Conference on Conceptual Modeling. Volume 3716 of Lecture Notes in Computer Science. (2005) 128–143.
17. Hoppenbrouwers, S., Proper, H., Weide, T.v.d.: Towards explicit strategies for modeling. In Halpin, T., Siau, K., Krogstie, J., eds.: Proceedings of the Workshop on Evaluating Modeling Methods for Systems Analysis and Design (EMMSAD'05), held in conjunctiun with the 17th Conference on Advanced Information Systems 2005 (CAiSE 2005), Porto, Portugal, FEUP, Porto, Portugal (2005) 485–492.