Towards Control Patterns for Smart Business Networks

Vera Kartseva, Joris Hulstijn, Ziv Baida, Jaap Gordijn, Yao-Hua Tan

vkartseva@feweb.vu.nl

Vrije Universiteit, Amsterdam

Abstract

To keep a smart business network sustainable, inter-organizational control measures are needed to detect or prevent opportunistic behavior of network participants. In this paper we present a methodology for understanding control problems and designing solutions, based on an economic value perspective. The methodology employs a library of so-called control patterns, inspired by design patterns in software engineering. A control pattern is considered as a generic solution for a common control problem. The adequacy and effectiveness of these control patterns is demonstrated by a case study about the redesign of customs procedures.

Keywords: governance and control, design patterns, business modeling, smart business networks

1. Introduction

Companies increasingly form Smart Business Networks (SBNs) to jointly satisfy a complex need. Well known examples include the networked business model of Cisco Systems [33] and the virtual integration of Dell Computers [22]. A *smart business network* is a group of enterprises with joint goals, linked by network technology, that collaborate and interact in such a way that the network remains sustainable and robust to defection of one of the actors, and allows each actor to increase its own value [35], p.229. To be sustainable, a network needs *control* measures, to prevent and detect opportunistic behavior of participants. A participant may for example leave the network prematurely, not fulfill its obligations, or commit fraud. Already from the early days of trade, transactions between organizations have therefore been governed by administrative control procedures, relying on a transfer of *control documents* [6]. While some control procedures are intended to govern commercial transactions between enterprises, governmental institutions like customs, have also introduced many control measures, to ensure for example the collection of taxes. Currently, the various governmental procedures around international trade are fragmented and implemented independently. Consequently, considerable overlaps exist between e.g., export, VAT, and excise procedures, as well as with existing commercial trading procedures. This means that a large number of different documents is required for each container crossing the border, and that there are many redundancies in business processes and information flows.

We present a case study, called the *Beer Living Lab*, about the redesign of customs procedures for the collection of excise duties. Governments, trading partners and technology providers collaborate in a network, to replace existing paper-based procedures, with procedures that rely on innovative technology. The purpose is to re-think why existing procedures work as they do, and how technological innovations can be used to tackle control problems in a network. In principle, all network partners should benefit. Customs administrations will achieve a higher degree of security and control. Trading partners increase their control of the supply chain, and by increasing credibility, they may also get a reduction in the required administrative procedures. Technology providers benefit from the research effort to provide open-source, secure, IT-based services. So the smartness of the collaboration lies in

the fact that innovative third party commercial IT-based services enable customs to achieve a higher degree of control, and provide a more competitive trading environment for business partners.

When designing or redesigning control procedures for a smart business network, it is crucial to guarantee that the new procedures provide the same safeguards as before, or better. To support the re-design of control procedures, a structured approach is therefore needed. For individual enterprises, approaches to analyze and design control procedures have been developed in the accounting and auditing fields, e.g. [32], [29]. But inter-organizational controls, as needed for a network of enterprises, have received only limited attention [6].

In this paper, we therefore propose the use of *control patterns*: a way to structure existing knowledge about the analysis and design of control procedures. Patterns generalize existing solutions for recurring problems, and make them accessible for re-use. They provide a structured way of encoding best practices. Such structured approaches are necessary, because domain experts often find it hard to make their knowledge explicit, and explain why a certain solution was chosen. The pattern approach has been proposed in architecture [1], and is very successful in software engineering [9]. Recently, patterns have also been applied to the business domain, for example in the design of administrative processes [26], organizational structure [8] and business process reengineering [4].

Based on a literature review, we have collected a *library of control patterns* for design and analysis of control procedures for inter-organizational settings. The adequacy and effectiveness of our library of control patterns is evaluated through a series of case studies, one of which is the Beer Living Lab. By applying the control patterns to the case description, we generate a *control model*, prescribing how controls should be designed. The control model is then used as a reference point, and compared with real life scenarios, provided by domain experts.

The control patterns are partly expressed using conceptual modeling techniques. Research on *business models* [3], [27], [34], [23], [28] utilizes conceptual modeling to provide concepts to judge and understand the viability of new business initiatives, which are often based on information technology. In the control patterns, we take two perspectives on conceptual modeling. Besides the *procedural* aspect of a control mechanism, we also provide a representation of the *business model* that motivates the execution of the controls. Our interpretation of the notion of 'business model' stresses the

importance of *economic value*. With *value modeling*, the focus is on *what* actors do and *why*, whereas process modeling concentrates on *how* they do it [11]. Business models proved to be a useful support tool in a setting in which decision making is done by multiple stakeholders [10]. Controls are typically not imposed by one central organization, but are negotiated among network participants. Stakeholders have different interests, and different views on the value proposition that underlies the network. This may lead to incomplete and ambiguous statements, when communicated in natural language [25]. The formal conceptualization of business ideas in a model, makes potential conflicts explicit, and may therefore help stakeholders resolve potential conflicts in an early stage of the development process.

The contribution of this paper is the following. Control patterns extend existing work on formal models of controls [6], [7], [20], by providing a structured approach to the (re)design of control mechanisms. The formal models of control focus more on checking the correctness of a formal specifications of a control mechanism. Furthermore, the library of control patterns covers a wider set of control mechanisms then those described by [6], [7], [20]. In addition to the customary procedural perspective, we also take a value perspective on the design of controls, which integrates control design into the *e*^{3}*-value* business modeling methodology.

The paper is organized as follows. Section 2 provides a theoretical background to our work, explaining the essentials of business modeling and control theory. Section 3 defines the notion of a control pattern and presents a library of control patterns for inter-organizational settings. Section 4 describes how we apply the control patterns in a case study about the redesign of customs procedures.

2. Theoretical background

A sustainable smart business network needs mechanisms for governing and controlling the collaboration and interaction between network participants. The ‘smartness’ of a network partly resides in the business model that underlies the network, which determines how the revenues from collaboration, such as efficiency gains, are redistributed among the participants. In most cases, such a business model is encoded in contractual arrangements between partners, and implemented through procedures and regulations. But contracts and regulations can be violated. In the context of control theory, a smart business network is therefore considered to be either in an *ideal situation*, in which no errors, oppor-

tunistic behavior or fraud occurs, or in a *sub-ideal situation*, in which some error, opportunistic behavior or fraud does occur [16]. Sub-ideal situations must be prevented, detected or corrected by means of a control mechanism. In the accounting literature [29], [32], ideal and sub-ideal situations are typically analyzed from an operational or procedural perspective, with process models and flow charts. In a business network, the ideal situation is often determined by the contractual arrangements that reflect the business model of the network. Therefore, we also need a value perspective to analyze the reasons for implementing a control mechanism.

In addition to this, there are other reasons for looking at control issues from a value perspective. First, the value perspective is conceptually close to Transaction Cost Economics, which studies contractual safeguards against opportunistic behavior in business relationships [37]. It facilitates a cost-benefit analysis of the control mechanism, which may also involve a risk assessment (e.g. [29], [32]). Second, control mechanisms are themselves services, with an additional price tag. That raises questions like: who is going to pay for a control mechanism, who is going to execute it, and how will it affect the individual business models of the parties involved? These questions are not particularly relevant from an internal control perspective, which is organized hierarchically, but in a business network controls may affect the profitability of participants, or may even lead to new business opportunities. Third, controls are often implemented with control documents, and some control documents have an inherent value aspect, and can for example be traded and resold (e.g. Bill of Lading, [6],[20]).

2.1 Business Modeling

There are several methodologies that address design issues of business models of network organizations, like BMO [27], value webs [33], and *e³-value* [10], [12]. Of these methodologies, *e³-value* is the only one that has a formal semantics, and that has a specific focus on value transfers between enterprises. The method is ontologically and formally well-founded, and is supported by graphical yet formal modeling software tools (see www.e3value.com). In this paper, we therefore apply the *e³-value* ontology for the description of so called ideal models [10], [12], to express organizations behaving in compliance with the trading procedures. Sub-ideal models are expressed using *e³-control*, a modification of the *e³-value* ontology, used to describe opportunistic behavior of actors [16].

2.1.1 Ideal value models

An *e³-value model* provides a conceptual model of the value transfers in a business network, encoded in the *e³-value* ontology [10], [12]. The *e³-value* constructs have a graphical notation. Figure 1(a) shows an example of a buyer who obtains goods from a seller and offers a payment in return. According to the law, the seller is obliged to pay value-added tax (VAT). This can be conceptualized by the following *e³-value* constructs (in bold). **Actors**, such as the buyer, seller, and the tax office are economically independent entities. Actors transfer **value objects** (payment, goods, VAT) by means of **value transfers**. For each value object, some actor should be willing to pay, which is shown by a **value interface**. A value interface models the *principle of economic reciprocity*: actors are only willing to transfer a value object, in return for some other value object. So only if you pay, can you obtain the goods and vice versa. A value interface consists of **value ports**, to represent that value objects are offered to and requested from the actor's environment. Actors may have a **consumer need**, which, following a **path of dependencies** will result in the transfer of value objects. Transfers are either dependent on other transfers, or lead to a **boundary element**. The *e³-value* methodology allows the designer to assign monetary values to value transfers and to calculate profitability of actors in a network.

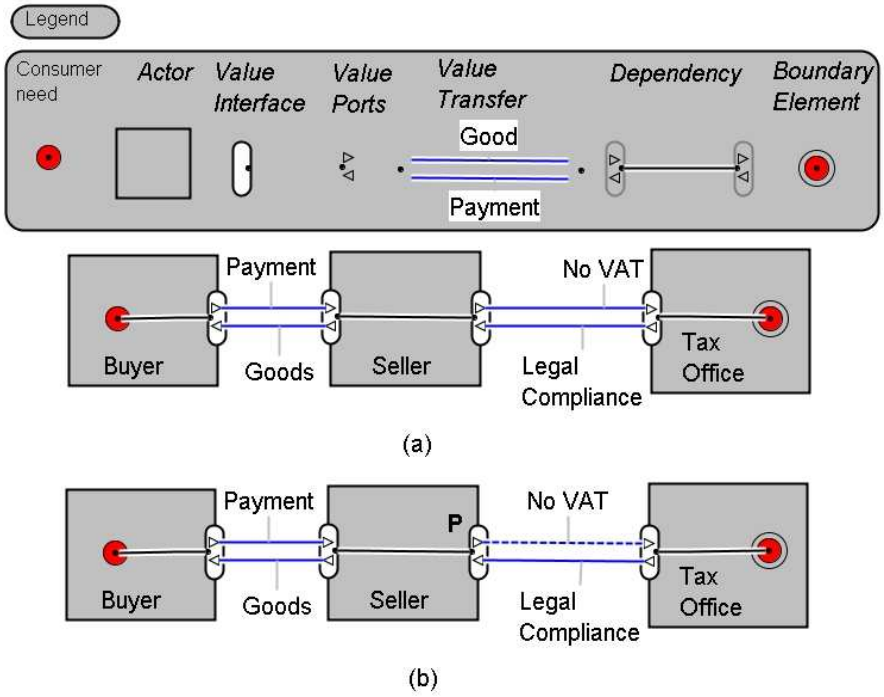

Figure 1. Example of an *e³-value model* of a purchase with tax payment;
(a) – ideal model, (b) – sub-ideal model

2.1.2 Sub-ideal value models

In *e³-value* it is assumed that actors behave in an ideal way, meaning that all value transfers occur as prescribed. This implies, among other things, that actors respect the principle of economic reciprocity. So, all the value ports in a value interface along a path of dependencies should transfer value objects, or none at all. But in reality, actors may not behave as represented in an *e³-value model*: they can commit fraud or make unintentional errors. In *e³-control*, these situations are modeled by **sub-ideal value transfers** [16]. These are graphically represented by dashed arrows, and can indicate different risks: e.g. actors will not pay for the goods, not obtain the goods, or obtain the wrong goods. For example, Figure 1(b) models a situation when the seller does not pay VAT tax. **P** is a **liability token**, assigned to the actor who is responsible for the sub-ideal value transfer. In this case that is the Seller.

2.1.3 Process models

Value models consider the transfer of valuable objects, like money, a good, or a service. However, in trading situations actors transfer more than value objects. For example, first a purchase order for a good is transferred, followed by a confirmation. Then, after a while, an invoice is sent, which is paid. Finally, a good is transferred. All these transfers require *operational* activities to be performed, by multiple actors, which can be shown using a business process model. By contrast, a value model only shows the transfer of value objects, which is effectively the transfer of ownership rights, see [17]. Additionally, business process models show the time-ordering of transfers of activities, whereas in value models we abstract from this ordering. We only show *that* objects are transferred, not in which *order*. The temporal order in which activities take place, forms a crucial part of the control mechanism. So in addition to value models we also need process models to capture control aspects. To represent the process aspects of control problems and their solutions, we employ UML-activity diagrams [30]

2.2 Control theory

To identify the problems and controls in an organization, a general assumption is that every activity in a process is a potential source of control problems. Control problems are typically identified by an analysis of risk indicators and threats discovered in an audit process. A *control mechanism* is a guideline on how to organize business processes in order to prevent, detect or reduce the risks posed by a control problem. This general framework is also relevant for inter-organizational controls [16].

Internal control theory is applicable to the design of inter-organizational controls only to some extent [6]. Internal control addresses control problems by organizational measures inside the organization. By contrast, in inter-organizational settings, control problems resulting from opportunistic behavior of partners in a network, are mostly dealt with by contractual arrangements [5]. Therefore, it is difficult to apply internal control guidelines directly to inter-organizational processes. In search for a formal theory of controls, we studied work of Chen [7] on *detective controls*, and Bons [5], Bons et al.[6] and Lee [20] about the design of *inter-organizational trade procedures*, which also involve preventative controls. We identify a vocabulary of terms, to be used in the definition of control patterns.

2.2.1 Detective Controls

Based on a review of internal control literature, Chen [7] pp. 62-67 presents a set of audit rules and principles[1], developed to be implemented in a decision support system. In such a system, risks are automatically detected by analyzing a business process and checking whether the audit principles have been applied or not. In Table 1 we present a summary of the audit principles.[2] The theory distinguishes between *operating activities* and *control activities*. In the context of detective controls, a control activity is interpreted as a kind of verification, which compares the results of the operating activity, with some claim or statement. Thus we use the following adapted vocabulary (in bold). A **verification activity** audits the results of an **operating activity** with respect to its legitimacy, quality

[1] Chen uses the term 'control pattern', but in a different sense: "a stereotypical description of the relationships between tasks, agents, assets, and information repositories involved in an internal accounting control system." [7] p. 16. This roughly corresponds to our notion of 'control mechanism'. In our patterns, a control mechanism acts as a solution, expressed by control principles which are based on Chen's work (see section 2.2.4).

[2] The presentation is based on Bons [5] p 55, and adapted for coherence with *e*^{3}*-value*. We use 'activity' instead of 'task', and 'verification' instead of 'control'. The numbering I –X is original.

or quantity [7]. The claims about the operating activity are represented by a **document to-be-verified**. Additional **supporting documents** represent evidence about the execution of the operating activity. For example, to verify whether the right type and quantity of goods were delivered, we use the purchase order as a supporting document. Such evidence is often produced by previous control activities.

Principles of precedence order of activities	
I.	Whenever an operating activity exists, a corresponding verification activity should also exist.
II.	Whenever an operating activity and its corresponding verification activity exist, the verification activity must always follow the operating activity.
Principles of relation between information and activities	
III.	When a verification activity exists, it must be furnished with supporting documents.
IV.	When a verification activity uses a supporting document, the supporting document should be verified by a previous control activity.
V.	A supporting document should be generated by a source independent of the source which generates the document to be verified.
VI.	If a control activity uses a supporting document, the supporting document should be transferred directly from the control activity, which verified it.
Principles of organizational structure (segregation of duties)	
VII.	A verification activity and the operating activity it intends to control should be segregated into two different positions.
VIII.	A verification activity and the operating activity it intends to control should be delegated to two different agents.
IX.	The position responsible for a verification activity must not be lower in the formal power hierarchy than the position of the operating activity to be controlled.
X.	The agent responsible for a verification activity should be socially detached from the agent responsible for the operating activity to be controlled

Table 1 Audit principles of Chen [7], adapted for readability and coherence of terminology

The audit principles are clustered as follows. Principle I and II deal with the *order* of activities. In detective control, the verification activity has to occur *after* the operating activity. Principle III - VI put additional requirements on the *supporting* documents. For example, Principle VI requires direct transfer: no intermediate parties should handle the supporting documents. This is crucial to avoid tampering, because in practice a very high percentage of fraud cases involves the alteration of otherwise valid documents. Principles VII - X are concerned with the organizational context and the *assignment* of activities to actors. These principles, among other things, ensure segregation of duties..

In this research we focus on inter-organizational transactions in a business network. Since the principles in Table 1 are originally developed for internal control, not all the principles are relevant. In particular, principles VII and VIII distinguish between positions and agents in an organization. But at the network level, we only model complete enterprises. Principle IX is not relevant, because at this stage of the research, we do not define hierarchical relationships between enterprises.

2.2.3 Inter-Organizational Controls

In inter-organizational settings, the typical scenario is that of a transaction between two parties. When parties do not have an existing business relationship, lack of trust is likely. Trust has been defined as "The willingness of a party to be vulnerable to the actions of another party based on the expectation that the other party will perform a particular action important to the trustor, irrespective of the ability to monitor or control that other party" [24], p 712. Without trust, a party that invests in a transaction, is uncertain whether the other party will perform its part of the deal, or behave opportunistically. This is called 'ex-post' opportunism [37]. The purpose of inter-organizational controls is to reduce the uncertainty, and provide enough guarantees for parties to engage in a transaction.

In our terminology, the risks are assessed from the point of view of the **primary actor**, who does not trust the **counter actor,** and must therefore design control mechanisms against sub-ideal behavior of the counter actor. From a value perspective, we can say that the primary actor transfers a **primary value object** (PO) to the counter actor, and the counter actor transfers a **counter value object** (CO) in return. From a process perspective, such a transaction is modeled by two activities. The primary actor performs a **primary activity**, and the counter actor executes a **counter activity**, each resulting in the corresponding value transfers. Figure 2 shows a value model of such a transaction on the left, and the corresponding process model on the right. The order in which the primary activity and counter activity occur is not specified. This is indicated by the UML notation for parallel execution (thick horizontal bar).

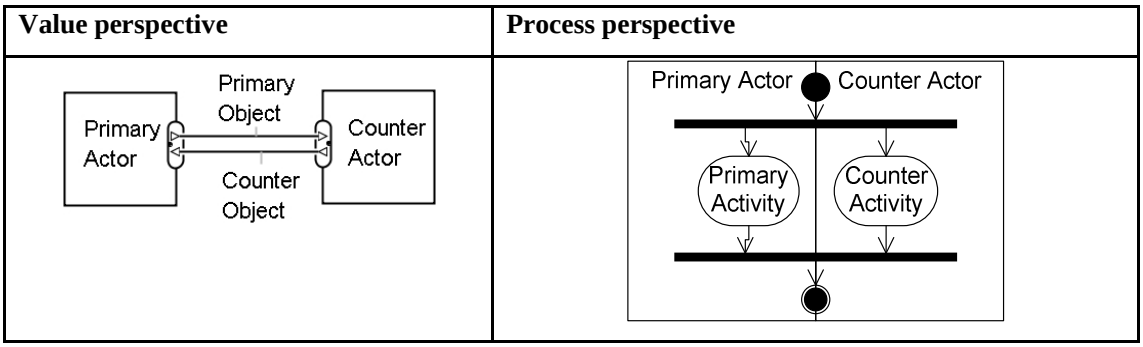

Figure 2.General transaction scenario

The control mechanisms considered by Bons [5], Bons et al.[6] and Lee [20], combine detective controls, as discussed above, with preventative measures. The mechanisms are described using

five General Principles, numbered GP I –V. Consider for example: "*General principle II: Before Role 1 executes a primary activity, it should have witnessed the performance of the counter activity by some Role 2, if the party playing Role 1 does not trust the party responsible for Role 2, unless it has received evidence that Role 2 has executed its tasks*", [5] p 61. Essentially, the counter activity must take place *before* the primary activity. Clearly, this will reduce the risks for the primary actor. However, any network participant can be primary actor or counter actor. For example, in a simple purchasing scenario, both the buyer and the seller can be seen as the primary actor, considering measures against opportunistic behavior. The seller will for example require a pre-payment arrangement, whereas the buyer requires delivery before payment. Clearly, this will lead to a conflict. Therefore, trade procedures have been designed, that provide guarantees to satisfy both participants. Consider for example a down payment, the use of an Escrow service [17] or a Bill of Lading [20].

A smart business network can be interpreted as a number of binary value transactions between actors. So we assume that all multi-party transactions can be decomposed into binary ones. For a proper network analysis, the network must be analyzed in a step-wise fashion, each time taking the perspective of one actor as primary actor, considering the risks of sub-ideal behavior of other actors.

We distinguish between *control mechanisms* and *control principles*. Literature on expert systems for internal control shows that experts typically apply deductive reasoning from *basic control principles* to more *specific control mechanisms* (see e.g. [21]). Control principles are basic rules, which cannot be further broken down. On the other hand, control mechanisms result from a combination of basic rules, applied to a specific control problem. Finally, *control patterns*, to be discussed in the next section, provide guidelines of when and how to apply a certain control mechanism.

In this section, we have discussed detective controls and inter-organizational trade procedures, based on preventative controls. But internal control theory also knows other kinds of controls. For example, the proper authorization of transactions [29], is neither covered by the auditing principles in Table 1, nor by the principles of Bons. The *library of control patterns*, presented in section 3 of this paper, extends the work of Chen [7] and Bons [5] by providing a structured approach, which can deal with a wider set of control problems. In addition, the pattern approach is meant to (re)design control mechanisms, not just to check correctness or trustworthiness of existing control mechanisms.

3. Control patterns

A pattern is a description of a general and accepted solution for some recurring problem. Many different definitions exist. Traditionally, a design pattern has four essential elements: *pattern name*, *problem*, *solution*, and *consequences* [9], p3. As we discussed in section 2, the design of control mechanisms requires a problem-solution structure, so we expect that a pattern approach provides a useful way to structure knowledge about controls. We have adapted the pattern definition to the domain of control theory. In particular, we have separated the description of the context in which a pattern is to be applied, from the problem which motivates the selection of a pattern (see also [8]). The context describes the business network with the actors, their relationships, such as the presence or absence of trust, and the activities that need to be controlled. The problem specifies a specific risk to be detected or prevented by the solution of the pattern. Following [8], we allow the specification of several variations of a solution, along with the conditions (forces) under which these variations apply. We have added a specific field for control principles, which provide a step-by-step guideline of how to apply the pattern, based on the theory discussed in section 2. We have also added a field of related patterns, taken from [9] p.7, to represent relations between patterns that affect their applicability. We do not include a separate field for the consequences and trade-offs of applying a pattern. In future work, in which we will investigate the costs and benefits of introducing a control mechanism and how this affects the business model of the network, consequences and trade-offs will play an important role.

3.1 Definition of a control pattern

A **control pattern** is a description of a generic and re-usable control mechanism for a recurring control problem, selected on the basis of the context of application. The structure of a pattern, called a *pattern template*, is the following:

name: a descriptive name of the pattern, used to select patterns from a pattern library.

context: a description of the business network to be controlled, modeled from an ideal perspective, meaning that no one behaves opportunistically. The context is represented by a value model (section 2.1.1), and if needed for understanding the context, also by a process model.

problem: a statement of one or more risks for opportunistic behavior. A control problem exists if there is some deviation of the prescribed transfers of economic value. So we model the problem by a sub-ideal value model (see section 2.1.2), using sub-ideal value transfers and liability tokens. Again, if needed, we also use a process model.

solution: description of a control mechanism, to detect, prevent or correct the control problem. The solution is described by both process models and value models, and motivated by *control principles* (see below). A solution may have different *variations*, along with *forces*, which are the conditions to select these variations.

related patterns: description of the relations of the current pattern with other patterns in the library.

control principles: a set of general constraints on the way a process must be performed. The principles fall into three groups, described in section 2.2.1.

3.2 Control patterns library

In this section we present a library of general control patterns (Table 2). To extract the patterns, we used the 'PattCaR method' from the patterns literature [31]. The followings steps were followed:

Step 1. Based on a literature review, we *identified* a preliminary set of control patterns. We used text books of internal control, [2], [14], [29], [32] and [5],[6],[20] about inter-organizational controls.

Step 2. For each identified pattern we collected a number of *examples*, from the literature as well as from previous case studies. The case studies were performed in different sectors, such as internet radio [15], renewable energy [16], international trade [19], and health care [18].

Step 3. The examples were *modeled* using activity diagrams, and value models, and described in terms of the control principles (see Table 1).

Step 4. A *commonality-variability analysis* [31] was performed to compare examples of each potential pattern. As a result we identified common concepts in the examples, such as activities, objects and control principles. This served as input for encoding the patterns in pattern templates, using a common vocabulary. Sections 3.3, 3.4 contain two examples. The commonality-variability analysis also produced the pattern Evidence Collection (see Table 2), which was not found in Step 1.

Step 5. We *validated* the patterns in case studies, one of which is presented in this paper.

Pattern	Risk of primary actor	Control by primary actor
Commitment Confirmation	counter actor may deny to have made a commitment to primary actor	require confirmation of commitment from counter actor, before executing primary activity
Commitment Authorization	counter actor may not be a reliable partner to make commitments with	before making a commitment, require authorization from an actor who verifies credentials of counter actor
Pre-execution	counter actor may not execute counter activity as agreed, but does get primary object	verify counter activity, before executing primary activity
Execution Confirmation	counter actor may claim that primary activity was not executed, and thus refuse to execute counter activity	require confirmation of execution of the primary activity , from counter actor
Verification	counter actor may not execute counter activity as agreed, but claim to have done it correctly	verify counter activity
Evidence Collection	a verification activity introduced by another pattern, lacks evidence	use trustworthy evidence documents to perform verification

Table 2 Library of Control patterns

Table 2 uses a specific vocabulary (underlined words), explained in section 2. Note that identification of a common vocabulary is also part of the PattCaR method [31]. Recall that the patterns are formulated from the perspective of a primary actor, who does not trust the counter actor.

We will now give a brief overview of the patterns. We illustrate the patterns by a simplified transaction scenario, in which a buyer (primary actor) has ordered some goods, and does not trust the seller (counter actor) to deliver. So in this case the primary activity is payment; the counter activity is delivery of the goods. After the overview, we present the patterns Verification and Evidence Collection in more detail, because they are used in the case study.

The **pre-execution** pattern simply requires the primary actor to verify execution of the counter activity, *before* executing the primary activity. This is based, among others, on Bons [6] GPs II-IV, discussed in section 2. The associated control problem for the primary actor is that otherwise, the counter actor may not execute the counter activity. In our example, the buyer will only pay for the goods, after having inspected that the right goods were delivered. When both actors apply this pattern, we get into conflict, and an intermediate solution like a down-payment must be applied [19].

The execution confirmation is described among others by Bons [6] in GP I. The associated control problem is that, in case of a dispute, the primary actor will not have independent evidence to prove that the primary activity was properly executed. The control therefore requires the counter actor

to provide documentary evidence of execution of the primary activity to the primary actor. Think of a receipt. In our example, the buyer will require a quittance from the seller, as evidence of payment.

The **commitment confirmation** is described among others by GP V of Bons [6] and Weigand and De Moor [36]. The associated control problem is that the counter actor may refuse to recognize that he made a commitment to the primary actor, and not execute the counter activity. The control requires the counter-actor to provide documentary evidence of the transaction commitment. Normally, this is done by getting the counter actor to sign a contract. In our example, the buyer will require a price quote or offer, that commits the seller to deliver at a certain price.

The **commitment authorization** pattern is related to a proper authorization of transactions and activities [29], [32]. The associated control problem, is a possible commitment to an unreliable counter actor. The controls require the primary actor to receive an authorization from a superior or trusted third party. The authorization is based on verification of properties of the counter actor, such as reputation or credit worthiness. In our example, the buyer can have an agency like the Chamber of Commerce check the credentials of the seller, before making any commitments.

The **verification** pattern is mainly based on the work of Chen [7] discussed in section 2. The associated control problem for the primary actor, is that the counter actor may not execute the counter activity, in the way that was agreed, while claiming to have done so. Therefore, the primary actor must verify, or have an external party verify the execution of the counter activity. A verification is interpreted as a comparison between the actual facts and some claim that was made. In our example, the buyer will inspect whether the right goods were delivered, according to the purchase order.

Finally, the **evidence collection** pattern puts additional constraints on the evidence needed in any verification activity. This pattern is also based on the work by Chen [7] discussed in section 2. The associated control problem, is that the verification activity lacks trustworthy evidence, to make an assessment. Therefore, the pattern requires that evidence for the verification activity is produced based on witnessing the verified activity.. In our example, witnessing corresponds to the buyer's inspection of the goods, directly after delivery.

The patterns relate to the different phases of the transaction cycle [36], see also [6], p30. The process of concluding a transaction consists of four phases: (1) the *preparation* phase, (2) the *negotia-*

tion phase, (3) the *execution* phase, and (4) the *acceptance* phase. The *preparation* phase is dealt with by Commitment Authorization. The result of the *negotiation* phase produces a Commitment Confirmation. The Execution Confirmation and Verification patterns deal with the *acceptance* phase. The Pre-Execution pattern covers the *execution phase*. Evidence Collection can occur as part of any phase.

Patterns are related with each other through their context. In particular, the pattern Evidence Collection assumes that a verification activity is already present in its context. A verification activity is introduced by the Verification, Pre-execution or Commitment Authorization patterns. So Evidence Collection must and can only be applied after these patterns.

Name: Verification
Context: Primary actor and counter actor transfer value objects, called primary object (PO) and counter object (CO). Primary actor does not trust counter actor.
Problem: Counter actor does not execute the counter activity as agreed, but claim to do so.
Solution: Primary actor must ensure that a verification activity is executed, after the counter activity. The verification activity controls the result of the counter activity. **Force *a***: Primary actor is able to verify the counter activity. **Variation *a***: The verification activity is executed by primary actor. **Force *b***: Primary actor is not able to verify the counter activity. **Variation *b***: The verification activity is delegated to a trusted third party (TTP), who must transfer a testifying document with the results of the verification to primary actor.
Related patterns: This pattern requires subsequent application of the Evidence Collection pattern.
Control Principles: **Principles on the order of activities** 1. Add Verify 2. Verify must be performed after counter activity **Principles of relations between information and activities** 3. A document to-be-verified (TBV Doc), is required as input for Verify. 4. Variation *b*: TTP sends a testifying document with the results of the verification. 5. Variation *b*: The testifying document must be transferred directly from TTP to primary actor **Principles of organizational structure** 6. Verify is performed by an actor, independent and socially detached from the counter actor. 7. TBV Doc is produced by the counter actor.

Table 3 Pattern "Verification"

3.3 Pattern "Verification"

The **Verification** pattern (Table 3) is based on the detective controls of Chen [7] in section 2.2. Verification is interpreted as a comparison between the results of the counter activity and some claim made about the results of the counter activity, the **document to-be-verified**. On the basis of a verifi-

cation, a decision is made: to accept the counter activity, or not. Control principles 1 and 2 in Table 3 require a verification activity to be executed after the controlled counter activity, which is based on Chen's auditing principles I and II, in Table 1. Principle 3 states that a document to-be-verified must be present, and principle 7 states that it should be generated by the counter actor, corresponding to auditing principle V. In addition, the verification activity can be delegated to a trusted third party (TTP), in case the primary actor cannot perform the verification itself. This is described in Variation *b*. In that case, principle 4 requires the TTP to send a **testifying document** to the primary actor, with the results of the verification. This is motivated by Bons [5] GP-III: *"If Role 1 cannot witness the performance of a counter-activity by some Role 2, then another Role 3 should testify the completion of Role 2's activity, if the party playing Role 2 is not trusted by the party playing Role 1"*. Principle 5 requires that the testifying document is transferred directly to the primary actor. This is based on auditing principle VI, of Table 1. Finally, principle 6 requires the verification to be performed by an actor who is independent and socially detached from the counter actor, which is based on the auditing principles VIII and X (segregation of duties).

Figure 3 contains a graphical version of the Verification pattern. Note the changes to the value model in Variation *b*, when the verification activity is delegated to a TTP.

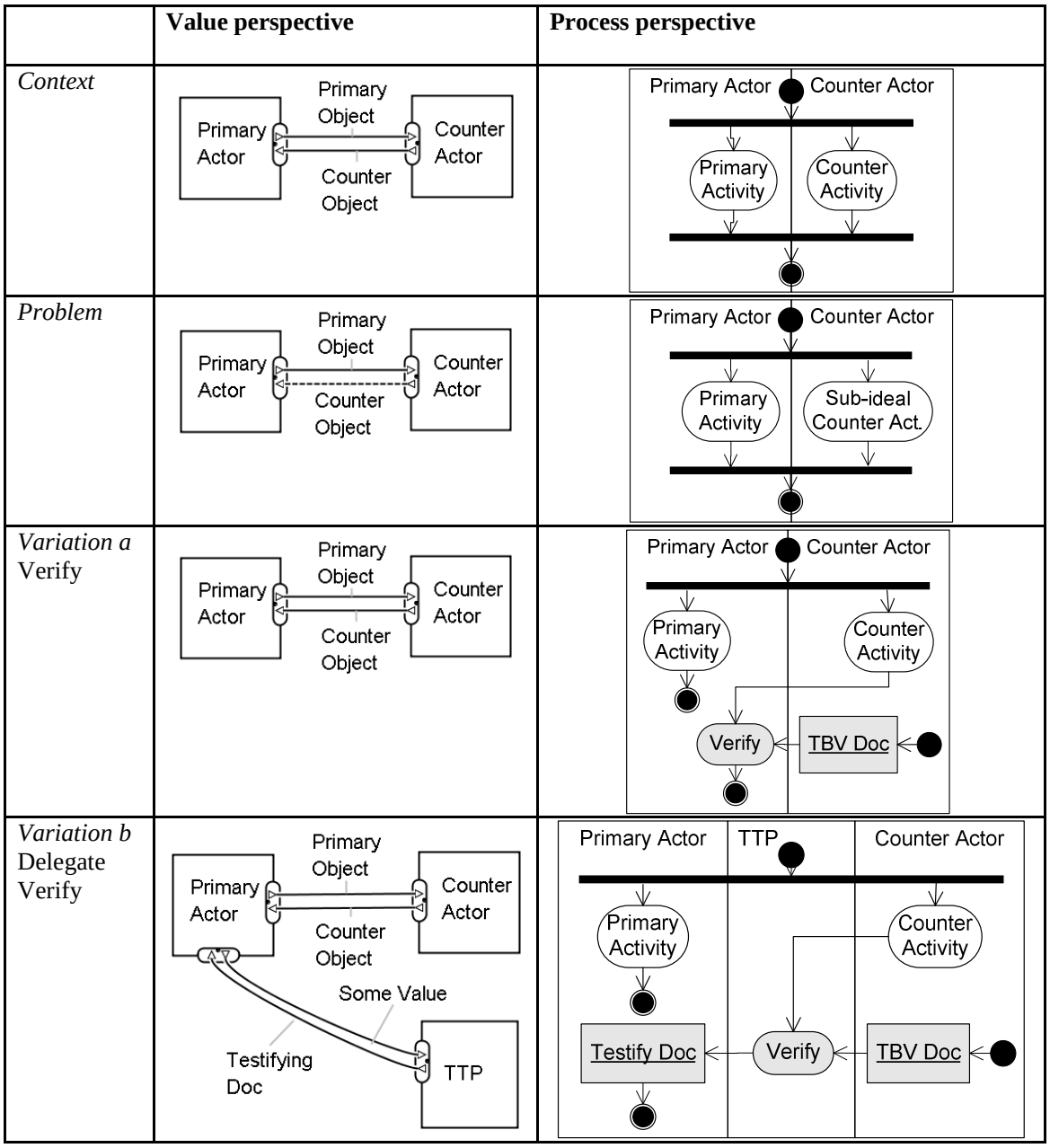

Figure 3 Verification Pattern

3.4 Pattern "Evidence Collection"

The Evidence Collection pattern puts additional requirements on the Verify activity, which is previously introduced by the patterns Verification, Pre-execution or Commitment Authorization. This is indicated in the context. The control problem is that the verification lacks evidence to make a decision. The solution of the pattern requires that verification is based on **supporting documents**, from some previous control activity. Here, we interpret this by *witnessing* (see also [5]). Witnessing is used

to collect evidence about the counter activity. Also other sources of evidence can be used. Variation *a* is applied when the primary actor or TTP who executes the verification activity is also able to witness the execution of the counter activity. Variation *b* is applied when witnessing must be delegated, because the actor has no direct access to the results of the counter activity. The trusted third party to whom the witness activity of the Evidence Collection pattern is delegated, is called TTP-E, to distinguish it from the TTP of the Verification pattern, which may be a different actor. The control principles 1, 2, 3 and 4 are loosely based on auditing principle III and IV of table 1. Principle 5 is based on auditing principle VI (direct transfer), and principle 6 on auditing principle V (segregation of duties).

Name: Evidence Collection
Context: The counter actor transfers a value object CO. A Verify activity, which verifies the counter activity, is executed by the primary actor or by a trusted third party (TTP).
Problem: The Verify activity lacks evidence and/or documents on which to base its assessment.
Solution A Witness activity is added before the Verify activity, to observe the outcome of the counter activity **Force** *a*: The primary actor or TTP has direct access to the outcome of the counter activity **Variation *a***: The Witness activity is performed by the primary actor or TTP. **Force *b***: The primary actor has no direct access to the outcome of the counter activity. **Variation b**: The Witness activity is delegated to (another) trusted third party (TTP-E).
Related patterns: This pattern puts additional requirements on the verification activity, previously introduced by the patterns Verification, Pre-execution or Commitment Authorization.
Control Principles: **Principles on the order of activities** 1. Add Witness activity. 2. Witness has to be executed before Verify. **Principles of relations between information and activities** 3. Variation *b*: Add supporting documents (Sup Doc) as an output of Witness. 4. Variation *b*: The supporting documents are input for Verify. 5. Variation *b*: The supporting documents must be sent *directly* to Verify. **Principles of organizational structure** 6. Witness is performed by an actor, independent and socially detached from the counter actor, with direct access to the outcome of the counter activity.

Table 4 Pattern "Evidence Collection"

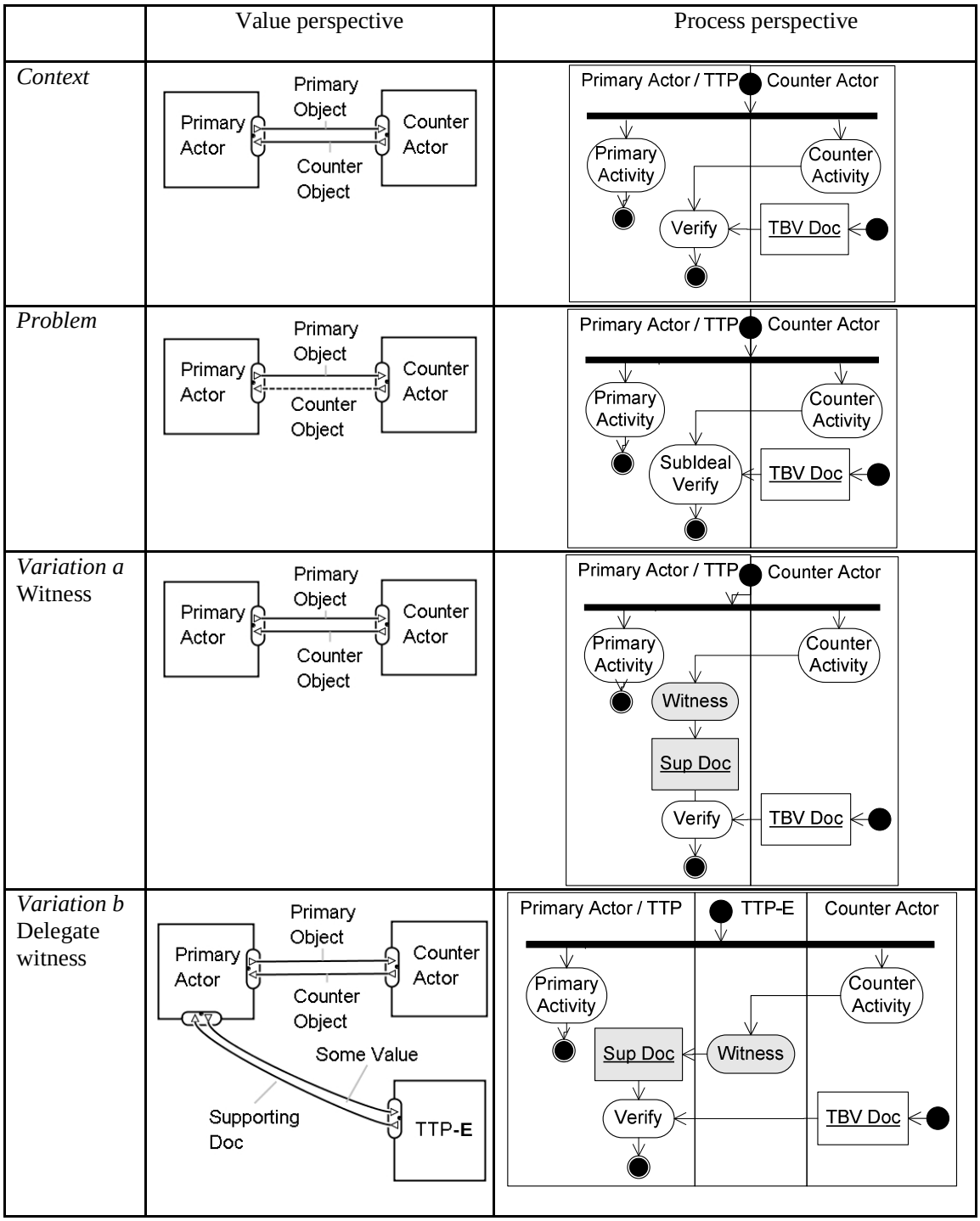

Figure 4. Evidence Collection Pattern

4. Case Study: Beer Living Lab

The use of the control patterns was studied in collaboration with domain experts from the Dutch Customs and Tax Administration, and experts from a large beer producer in the Netherlands. Hence, the Beer Living Lab can be categorized as an observational case study [13].

The study was designed as follows. First, an initial analysis of the case was made, on the basis of interviews with domain experts and existing documents. We identified the various actors involved, and the objects of value they transfer, and we made a partial analysis of the business processes regarding excise duties. Second, we identified control problems. Third, we applied the solution indicated by the patterns. This produced a so called *control model*: a model of the control mechanisms that ought to be implemented. Fourth, this control model was compared with a number of real life scenarios, provided by the domain experts. The comparison was meant to validate our interpretation of the case description and our application of the control patterns. Moreover, as part of the redesign effort, we wanted to find out whether the real life scenarios needed to be redesigned at all. Note that control patterns are *prescriptive*: they do not describe facts, but prescribe how control mechanisms must be designed. That means that there are two ways in which a real life scenario can deviate from the control model: either (1) the model is based on a wrong or incomplete interpretation of the relevant facts, or (2) reality does not comply with the model, which means that we found a control problem.

To apply the patterns, we take three steps [16]. First, we make an ideal model of the case, using value and process models. Second, we identify a control problem, by modeling a sub-ideal value model. Third, we match each identified control problem with the control problems listed in the pattern library in Table 2 and select a pattern, which deals with this problem. We apply the solution of the pattern to our sub-ideal model, and adapt the value model correspondingly.

4.2 Case Description

When goods like beer and cigarettes, called excise goods, are sold, the seller needs to pay a special tax called excise. The general principle is that excise duties only have to be paid in the country in which the excise good is sold and consumed. Hence, if a beer producer in the Netherlands, say BeerCo NL, is exporting beer to a retailer in the United Kingdom, possibly through an associate BeerCo UK, ex-

cise has to be paid by the British retailer to Customs UK[3]. In this case, the beer producer in the Netherlands can export excise-free. Clearly, this is only acceptable for the Dutch Customs and Tax Administration, if the beer producer in the Netherlands can prove that the goods were indeed shipped abroad. The procedures currently revolve around the transfer of paper documents. The core document for this export procedure is the Administrative Accompanying Document (AAD). This document is signed by a so-called Excise Warehouse (EW) in the UK. Customs UK subsequently signs the AAD, to confirm that the goods did indeed arrive in the UK. Finally, the AAD is returned to the Dutch beer producer as evidence, to be presented to the Dutch Customs and Tax Administration upon request. In this paper we specifically look at the control problems of the AAD procedure. We analyze the AAD procedure and indicate how the procedure can be replaced by smart technology.

4.3 Ideal value model

When BeerCo NL exports beer to the UK, it is exempt from excise duties and is considered compliant with the law. This is represented in Figure 5 by the transfer between BeerCo NL and Customs NL, which is linked by a dependency path to the export exchange with BeerCo UK. BeerCo UK sells the beer to a Retailer with EW. Such a retailer is officially licensed for excise handling. The retailer with EW sells the beer on the UK market, for a price that covers the excise, and pays the required excise duties to Customs UK. Figure 6 shows the corresponding process model. In the remainder of the case study, we concentrate on the black part of the process model, about the AAD procedure.

4.4 Sub-ideal value model

The control problem solved by the AAD procedure, is that a certain amount of beer is sold abroad (excise free), which is in fact sold in the Netherlands (not excise free). This problem is modeled in Figure 7. BeerCo NL delivers beer to consumers in the Netherlands. However, the OR-fork (triangle) models that BeerCo NL has a choice. It can pay excise for the beer sold in the Netherlands (Excise

[3] In some countries excise is considered a tax issue, while in other countries it is considered a customs matter. We therefore refer to Tax and Customs organizations interchangeably.

NL) in return for legal compliance (LC), or it can declare that the beer was as exported (incorrect ED) and still get an excise exemption and legal compliance (LC NL), for beer that did not cross the border.

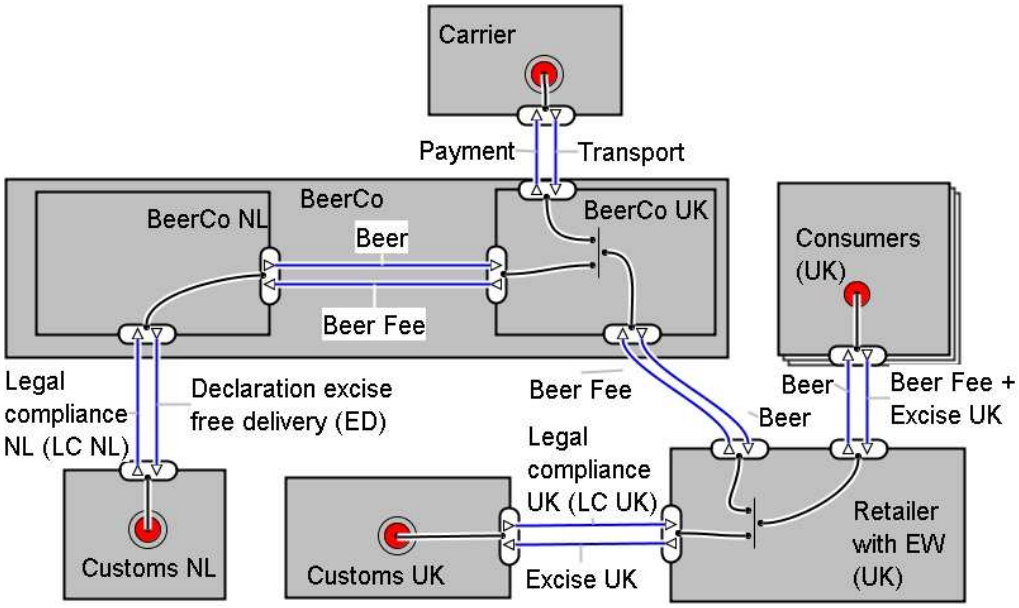

Figure 5. Ideal business model for beer export

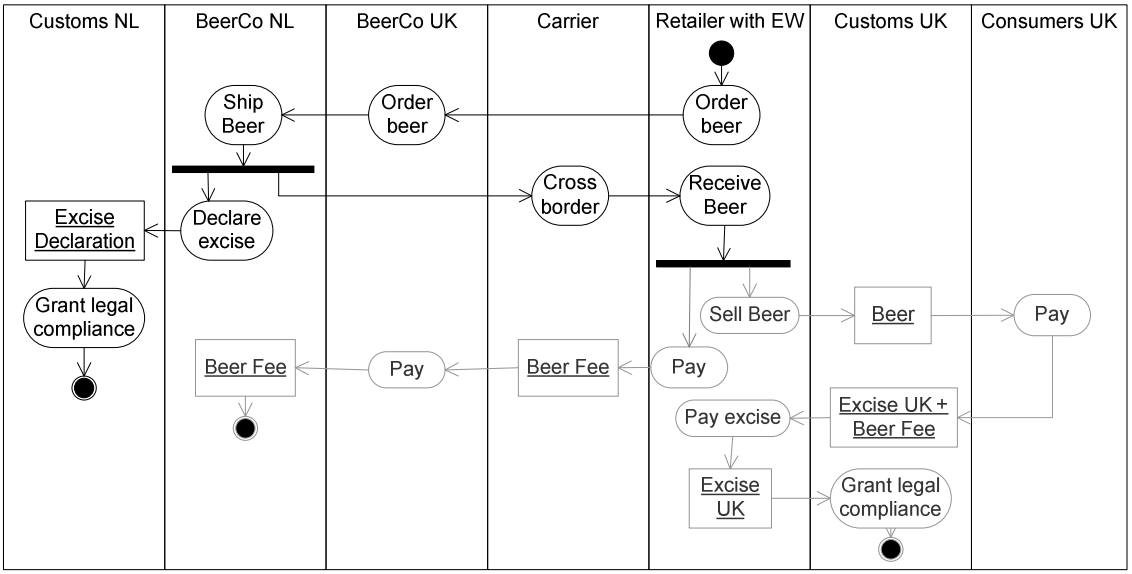

Figure 6. Corresponding (partial) process model for beer export

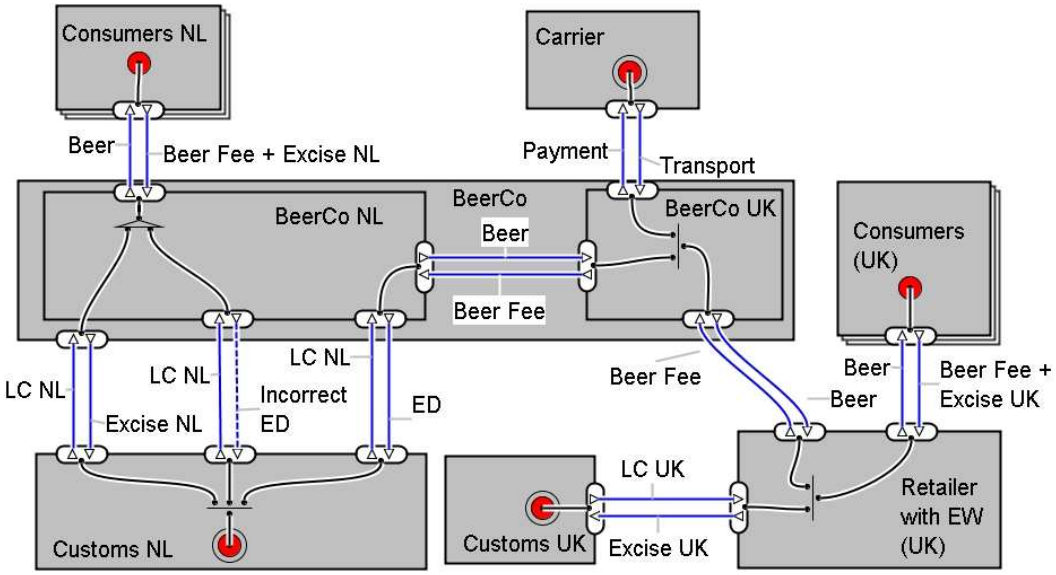

Figure 7. Sub-ideal value model for beer export from The Netherlands to the UK

4.5 Selection of a pattern

Now we match the sub-ideal model to the patterns. We do this by assigning actual actors and activities to the variables primary actor, counter actor etc. Here Customs NL is primary actor, because they do not trust counter actor BeerCo NL. Grant Legal Compliance is primary activity. Cross Border is taken as counter activity, because it provides the material facts, which give right to excise-free sales; Excise Declaration is only a procedural step. We substitute the assignments into the context and control problem of each pattern in the library (Table 2) and select the pattern that matches best (Table5).

Pattern	**Risk of primary actor (Customs NL)**	**Fit**
Commitment Conf.	BeerCo NL may deny to have made a commitment to Customs NL	○
Commitment Auth.	BeerCo NL may not be a reliable partner to make commitments with	○
Pre-execution	BeerCo NL does not execute Cross Border, but gets Legal Compliance	?
Execution Confirmation	BeerCo NL claims that Grant Legal Compliance was not executed and refuse to execute Cross Border	○
Verification	BeerCo NL does not execute Cross Border as agreed, but claim to do so	•
Evidence Coll.	Not applicable; no Verify activity in context	○

Table 5. Pattern Selection

We found that the control problem that underlies the AAD procedure most resembles the control problem of the Verification pattern: "BeerCo NL does not execute Cross Border as agreed, but claim

to do so". More precisely, some of the beer that is stated on the Excise Declaration, does not cross the border. The Verification pattern requires verification of every Excise Declaration, which is taken here as the document to be verified. The initial model in Figure 6 does not have such a verification activity.

The Pre-execution pattern could also be a candidate. Pre-execution would solve the problem that Customs NL grants legal compliance (excise exemption), while the corresponding beer has not crossed the border. But in this business-to-government setting, such a situation is not a real problem. When Customs NL would later discover a violation of the excise laws, they have the institutional power to withdraw the legal compliance from, and reclaim the lost excise duties. The importance lies in the verification, which is also part of the Pre-Execution pattern, not in the order of activities.

4.6 Control Model

Now we apply the selected pattern Verification to the case. Pattern Verification has two variations. Variation *a* is applied when the primary actor can perform the verification. Because Customs NL is indeed able to verify the Excise Declaration, we apply Variation *a*. According to the control principles in Table 3, the following guidelines are observed:

1. Activity Verify is added.
2. Verify is performed after Cross Border.
3. Verify takes Excise Declaration (document to be verified) as input.
4. Not applicable (variation *b*).
5. Not applicable (variation *b*).
6. Verify is assigned to Customs NL (primary actor).
7. Excise Declaration (document to be verified) is generated by BeerCo NL (counter actor).

After adding the Verify activity, the sub-ideal model has changed. The adapted sub-ideal model is again matched against the context and control problems of the patterns in the library. In the current sub-ideal model, the Verify activity lacks evidence. So we select the Evidence Collection pattern. There are two variations. Variation *a* is applied when the primary actor has direct access to the outcome of the counter activity. Otherwise, Variation *b* is applied. The most elegant solution would be Variation *a*, in which witnessing the Cross Border activity is done by Customs NL. But since offi-

cially there are no longer any borders between EU member states, and goods can travel freely without reporting to customs, Customs NL cannot witness how much beer is crossing the border. So, we choose Variation *b*, where Customs NL must rely on documentary evidence from some other party confirming export. According to the control principles in Table 4, the following steps are executed:

1. Activity Witness is added.
2. Witness is performed before Verify.
3. Witness produces Supporting Documents as output.
4. Supporting Documents are input to Verify.
5. Supporting Documents must be sent directly to Verify.
6. Witness is performed by a hypothetical actor TTP-E, independent and socially detached from BeerCo NL (counter actor), and with direct access to the outcome of Cross Border.

Figure 8 shows the result of applying both patterns to the original model in Figure 6 [4]. The model in Figure 8 is called the *control model*. Many actors can fill the role of TTP-E, and supply the supporting documents. Thus, the control model acts as a kind of functional specification that can be implemented in different ways, by specifying which actor performs the TTP-E's activities. Note that the role of Retailer with EW can be played by any Retailer. No special status is necessary in this control model.

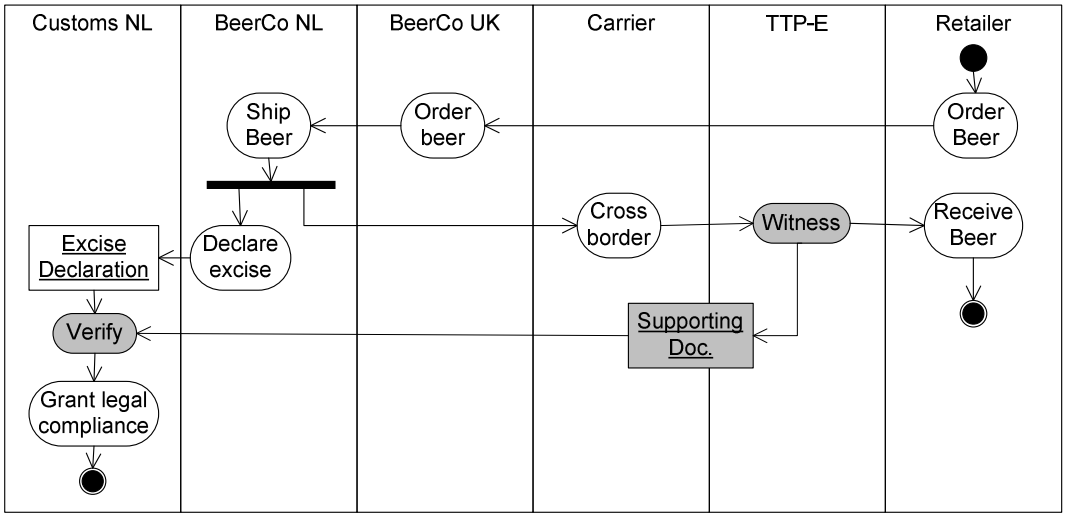

Figure 8 Control model derived from the patterns

[4] For brevity Figure 8 only contains the part of 0 about the controls. The grey part is not included.

4.7 Real-world scenarios

In this section we examine current and future export practices, and compare them to the control model. We analyze the current AAD procedure, shown in Figure 9. In this process the role of the TTP-E, to provide supporting documents, is performed by a Retailer with EW. In fact, Customs NL delegates the witnessing of export to Customs UK, who further delegates it to a Retailer with EW. Not every retailer can validate the AAD; only retailers with a special accreditation for the Excise Warehouse function. Accreditation procedures are covered by the pattern Commitment Authorization.

Another difference between the control model (Figure 8) and the current practice (Figure 9) is that in reality the AAD is not transferred directly to Customs NL. The AAD is transferred first to a carrier, then to BeerCo NL, and finally to Customs NL. This indirect transfer violates control principle 5 of pattern Evidence Collection, that supporting documents should be transferred directly, to prevent manipulation. Only if a supporting document cannot be forged, which is not the case here, indirect transfer is acceptable. According to the domain experts, this diversion from the control model indicates a real and existing control problem.

A further difference is that verification of Excise Declarations does not take place on a 100% basis. Only random checks can be performed, which shows another control weakness.

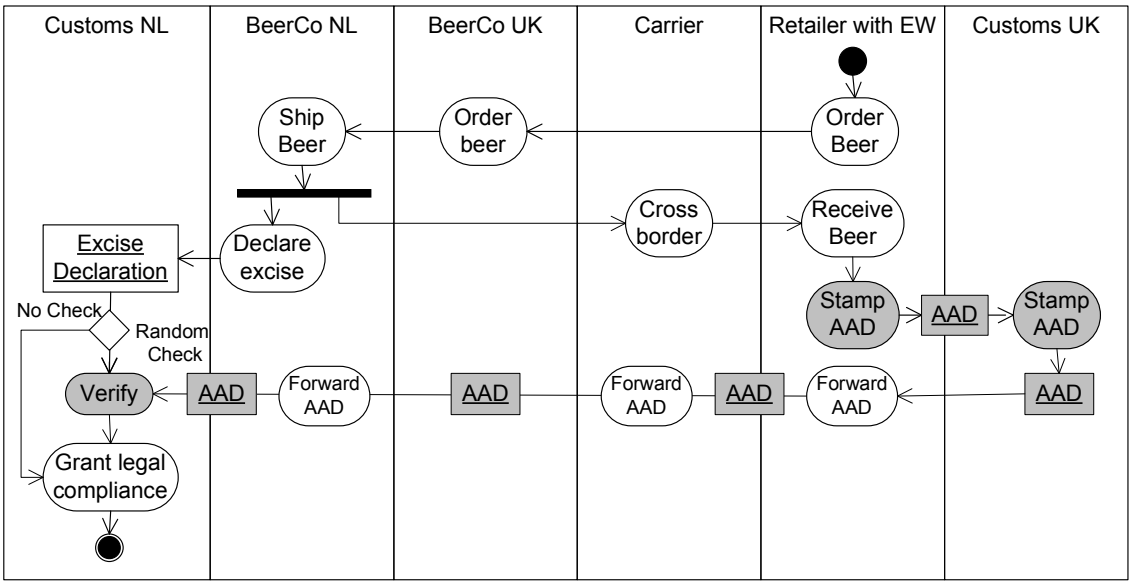

Figure 9. Current practice with AAD

We have also analyzed other real-life scenarios, including export to the USA, where export procedures are different from the procedures for cross-border deliveries inside the EU. We also investigated a scenario that uses a newly developed EU-wide information system, called Excise Movement Control System (EMCS). For a large part, these scenarios proved to be instantiations of our control model. But just like in the AAD case, we were able to identify remaining risks, by comparing scenarios to the control model and validating the deviations that we found with domain experts.

The control problem of the AAD can be solved by advanced technology, in particular, by a Tamper-Resistant Embedded Controller (TREC[5]), which is currently being developed. A TREC device can detect whether a container is opened by an authorized or non-authorized person, and send a message in case of tampering. TREC is intended to reduce fraud and increase security. Because the container's location is monitored through GPS technology, the device can detect when the container leaves the Netherlands. As an additional benefit, a TREC device could therefore replace the AAD. We propose a possible implementation in Figure 10. This scenario is an improvement over the current practice, because the TREC device can send a message directly to Customs NL, at the exact moment when the container crosses the border. By contrast, an AAD often returns only after three months.

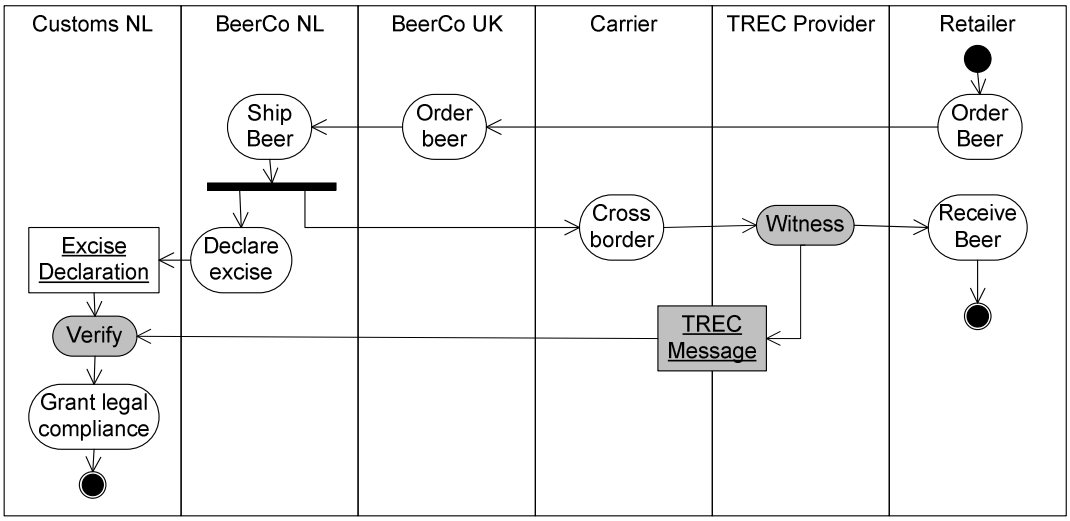

Figure 10 Export of beer using a TREC device

[5]Further information on TREC is available at http://domino.research.ibm.com/odis/odis.nsf/pages/board.06.html

Figure 10 is another instantiation of the control model. The TREC device performs the Witness activity, and therefore the organization that operates the TREC system, the TREC provider, takes the role of TTP-E. A pre-requisite is that the TREC provider is independent and socially detached from BeerCo NL on the company level and on the employee level (control principle 7, Table 3). To verify this, customs organizations will have to carefully assess the reliability of the technology providers.

4.8 Lessons Learned

Several lessons can be learned from the application of the control patterns to the case study, both about the content of the case, and about the adequacy and effectiveness of the patterns. Regarding the redesign of customs procedures for excise duties, we can conclude that the current practice is vulnerable. The AAD, which should prove that goods have indeed left the country, is transferred along all the parties of the supply chain, which makes it vulnerable to tampering. This control problem can be mitigated in several ways. One of these would involve a technical device, the TREC, and a new actor, the TREC provider. The TREC device will send an electronic message, when the goods have crossed the border. The message is delivered directly to the Dutch customs office. The export procedure that involves TREC is currently being designed as an instantiation of the control model generated by the control patterns.

The challenge of TREC lies in developing a reliable mechanism, to determine which parties are authorized to access a container, at which point in time. This authorization mechanism would require some form of information sharing between the customs' information systems, and the information systems of participants in a trading process. That means that parties must be carefully scrutinized, before they can join. A similar certification practice is now being considered, for other customs and tax procedures too. Under this scheme, some privileged companies, which can prove that they have excellent internal control, can reach the status of Authorized Economic Operator (AEO). AEOs will be exempted from some administrative procedures, but in return, the customs office has the right to access the AEO's information systems at all times. The purpose is to reduce the administrative burden, both for the customs office and the AEO.

One could say that such certification practices provide an example of the creation of a smart business network: efficiency gains are made, in return for the right to access information. This corresponds to other well-known cases of smart business networks, such as Dell Computers [22], where efficiency gains are made by sharing information along the supply chain. In general, before parties are willing to give other network participants access to their information, they need guarantees against opportunistic behavior. Such guarantees are covered by the Commitment Authorization pattern discussed in section 3. In future work, we will investigate certification and this pattern in more detail.

Regarding the use of control patterns, we can say that it has been successful. The patterns have been applied to a non-trivial case study, producing interesting results. In general, when designing an artifact, one can observe a trade-off between the adequacy and generality of a design guideline. On the one hand, a guideline must be generalized to different contexts, but on the other hand, if the guideline is too general, it becomes irrelevant to the immediate concerns of the designer. The same could be true for our control patterns, but the case study demonstrates that the control patterns in our library are not too general: they helped to reveal real control problems, and suggest a viable solution. Provided that a library of patterns is adequate for an application domain, a designer does not have to consider other potential solutions. So the use of patterns, will reduce the design space: the space of possible solutions, out of which a designer makes a choice.

5. Conclusions

The smartness of a smart business network may lie in the way in which the revenues or efficiency gains resulting from collaboration, are redistributed among participants. Participants who invest in the network depend on each other to receive these benefits, and are therefore vulnerable to opportunistic behavior. So in order to be sustainable, a smart business network needs to address governance and control. However, existing approaches for the analysis and design of business models, do not address control issues. A business model is typically created as an ideal situation, without considering the risk that some participants may fail to live up to expectations, or even commit fraud. Such opportunistic behavior can in fact be detected or prevented, by incorporating explicit control mechanisms into the inter-organizational business processes which regulate transactions between parties in a network. In

this paper, we have provided a structured approach, to the design and redesign of control mechanisms in smart business networks. In particular, we propose the use of *control patterns*, which structure existing knowledge about inter-organizational controls. The primary intent of a pattern is to provide a useful abstraction of an existing solution to a recurring problem, for the sake of reuse.

From a theoretical perspective, control patterns are innovative in several ways. First, in comparison with textbooks in accounting [2], [14], [21], [29], [32], the control patterns provide a structured way of designing controls, using conceptual modeling techniques well-known from software engineering. Second, the library of control patterns that we present, is specifically tailored for inter-organizational control problems, dealing with the risks of conducting a transaction between parties that may not trust each other. Third, our contribution extends existing formal models of inter-organizational control [6],[7],[20], on which some of the patterns are based, by providing specific guidelines to the design of controls, which can deal with a wider set of control problems. Fourth, in addition to the customary procedural perspective, we take a value perspective, modeling business networks by the transfer of economic value between participants. For this purpose we use the *e^3-value* methodology. One of the objectives of our approach, is to allow designers to evaluate the impact of the introduction of a new control mechanisms, on the business model that underlies a smart business networks. This will involve a cost-benefit analysis of applying a specific pattern, and investigate the introduction of new actors, or new business opportunities.

We demonstrate how to put theory into practice by a large scale case study about the redesign of customs procedures for excise duties: the Beer Living Lab. By applying control patterns we created a control model, used as a reference point. The case study revealed that current EU practice is vulnerable to fraud with excise declarations. Furthermore, the control model is used in a project in which businesses and governments participate to redesign export procedures, enabled by advanced technology. The procedure is being redesigned according to the control patterns. Domain experts from the participating organizations confirmed that the control model, which is based on application of the control patterns, does manage to identify real control problems, and suggest a viable solution.

One of our scenarios introduces a new actor to the business network: the TREC provider. That means that the business model of the network will be changed, and that the financial feasibility of

possible new business models must be closely examined. Such a profitability analysis is made possible by the *e*^{3}*-value* methodology. In future research, we will investigate various business models for operating the technology, that enables redesigned customs procedures.

The network of trading partners, customs administrations and technology providers involved in the redesign effort, is a typical example of a Smart Business network. The consortium can be considered smart in two ways. First, it uses advanced technologies to replace paper-based procedures to increase security, while reducing the administrative burden. Second, customs organizations rely on third party commercial services, instead of implementing and enforcing control measures by themselves. This practice is not new for businesses, but it is relatively new for governmental organizations.

Acknowledgements

The research of the first author is funded by the Post Master EDP Audit Education of the Vrije Universiteit Amsterdam. Part of the research was funded by The Freeband FRuX project. This research is part of the ITAIDE project. ITAIDE (Information Technology for Adoption and Intelligent Design for E-government) project (nr. 027829) is funded by the 6th Framework Information Society Technology (IST) Program of the European Commission, see www.itaide.org. We are greatly indebted to other participants of these projects for their valuable contributions.

References

[1] Alexander, C. (1979). The Timeless Way of Building. Oxford, Oxford University Press.

[2] Arens, A.A., Loebbecke, J.K. (1997) Auditing. Prentice Hall; 7th Revised edition

[3] Baida, Z., Gordijn, J., Akkermans, H., Sæle, H. and Morch, A.Z. (2005). Finding e-Service Offerings by Computer-supported Customer Need Reasoning, International Journal of E-Business Research 1(3): 91-112.

[4] Beedle, M. (1997) Pattern Based Reengineering. Object Magazine, January.

[5] Bons, R.W.H. (1997) Designing Trustworthy Trade Procedures for Open Electronic Commerce, PhD Thesis, EURIDIS, University of Rotterdam.

[6] Bons, R. W., Lee, R.M., and Wagenaar, R.W. (1998). Designing Trustworthy Inter-Organizational Trade Procedures for Open Electronic Commerce. International Journal of Electronic Commerce, 2(3):61–83.

[7] Chen, K. (1992). Schematic Evaluation of Internal Accounting Control Systems. PhD thesis, University of Texas at Austin, revised version available as Chen, K. and Lee, R.M. (1992), EURIDIS Research Monograph RM-1992-08-1.

[8] Coplien, J. O., Harrison, N.B. (2004) Organizational Patterns of Agile Software Development, Prentice Hall

[9] Gamma, E., Helm, R., Johnson, R., and Vlissides, J. (1995). Design Patterns: Elements of Reusable Object-Oriented Software. Addison Wesley, Boston.

[10] Gordijn, J. and Akkermans, J. (2001). e3-value: Design and evaluation of e-business models. IEEE Intelligent Systems, Special Issue on e-Business, 16(4):11–17.

[11] Gordijn, J., Akkermans J.M., and Van Vliet J.C. (2000), "Business Modelling is not Process Modelling". in: Conceptual Modeling for E-Business and the Web, LNCS 1921, pg 40-51 .

[12] Gordijn, J. and Akkermans, J. (2003). Value-based requirements engineering: Exploring innovative e-commerce ideas. Requirements Engineering, 8(2):114–134

[13] Hevner, A.R., Ram, S., and March, S.T. (2004). Design science in information systems research. Management Information Sytems Quarterly, 28(1):75–105.

[14] Hollander, A.S., Denna, E., Cherrington, J.O. (1999) Accounting, Information Technology, and Business Solutions. 2nd ed McGraw-Hill

[15] Kartseva, V., Gordijn, J., and Tan, Y.-H. (2005a). Designing control mechanisms for value webs: The internet radio case study. Proc. of the 18th Bled Conference: eIntegration in Action(Bled'05)

[16] Kartseva, V., Gordijn, J., and Tan, Y.-H. (2005b). Towards a modelling tool for designing control mechanisms in network organisations. International Journal of Electronic Commerce 10(2):57–84.

[17] Kartseva, V., Gordijn, J., and Tan, Y.-H (2006) Inter-Organisational Controls as Value Objects in Network Organisations", In: Dubois, E. and Pohl, K. (eds.) Proceedings of the 18th Conference on Advanced information Systems Engineering (CAISE'06)

[18] Kartseva, V. and Tan, Y.-H. (2005). Designing controls for a marketplace of health care services: a case study. In Proceedings of the 12th Research Symposium on Emerging Electronic Markets (RSEEM 2005). Vrije Universiteit, Amsterdam

[19] Kartseva, V., Hulstijn, J., Gordijn, J., and Tan, Y.-H. (2006). Towards Value-based Design Patterns for Inter-Organizational Control. In Proc. of the 19th Bled Conference: eValues (Bled'06).

[20] Lee, R.M. (2002) Automated Generation of Electronic Procedures: Procedure Constraint Grammars, Decision Support Systems 33: 291 – 308.

[21] Looi, C.-K., Tan, S.L., Teow, P.C., and Chan, H.S. (1989). A knowledge-based approach for internal control evaluation. Proceedings of the 2nd international conference on Industrial and engineering applications of artificial intelligence and expert systems, 254–261.

[22] Magretta, J. (1998) The power of Virtual Integration: An Interview with Dell Computer's Michael Dell, Harvard Business Review, 2(76):72-84.

[23] Malone, T.W., Crowston, K., Pentland, B., Dellarocas, C., Wyner, G., Quimby, J., Osborn, C.S., Bernstein, A., Herman, G., Klein, M., O'Donnel, E. (1999). Tools for Inventing Organizations: Towards a Handbook of Organizational Processes. Management Science, 45(3): 425—433.

[24] Mayer, R., Davis, J., and Schoorman, F. An integrative model of organizational trust. Academy of Management Review, 20(3):709–734, 1995.

[25] Meyer B. (1985) On Formalism in Specifications. IEEE Software, (2)1: 6 -26.

[26] Motschnig-Pitrik, R., Randa, P., and Vinek, G. (2002). Specifying and analysing static and dynamic patterns of administrative processes. In Proceedings of the 10th European Conference on Information Systems (ECIS 2002), Gdansk, Poland.

[27] Osterwalder, A. (2004), The Business Model Ontology: A Proposition in a Design Science Approach, PhD thesis, University of Lausanne, Lausanne, Switzerland.

[28] Pateli, A.G., Giaglis, G.M. (2004) A Research Framework for Analysing Business Models. European Journal of Information Systems, 13(4):302-304.

[29] Ronmey, M. and Steinbart, P. (2003). Accounting Information Systems. Prentice Hall, New Jersey, 9th edition.

[30] Rumbaugh, J., Jacobson, I., and Booch, G. (1999). The Unified Modelling LanguageReference Manual. Addison Wesley, Reading, MA.

[31] Seruca, I. and Loucopoulos, P. (2003). Towards a systematic approach to the capture of patterns within a business domain. The Journal of Systems and Software, (67):1–18.

[32] Starreveld, R., de Mare, B., and Joels, E. (1994). Bestuurlijke Informatieverzorging (in Dutch), volume 1. Samsom, Alphen aan den Rijn, 4th edition.

[33] Tapscott, D., Lowy, A. and Ticoll, D. (2000): Harnessing the Power of Business Webs, Harvard Business School Press, Boston

[34] Tillquist, J., King, J., and Woo, C. (2002) A Representational Scheme for Analyzing Information Technology and Organization Dependency, Management Information Systems Quarterly, 26(2): 91-118.

[35] Vervest, P., Preiss, K., van Heck, E., and Pau, L.-F. (2004). Introduction to smart business networks. Journal of Information Technology, 19: 228–233.

[36] Weigand, H., & Moor, A. de (2003). Workflow analysis with communication norms. Data and Knowledge Engineering, 47(3): 349-369.

[37] Williamson, O. E. (1979). Transaction cost economics: The governance of contractual relations. Journal of Law and Economics, 22:3–61.