19th Bled Electronic Commerce Conference

eValues

Bled, Slovenia, June 2006

Towards Value-based Design Patterns for Inter-Organizational Control

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

Vrije Universiteit, Amsterdam

{vkartseva, jhulstijn, ytan}@feweb.vu.nl, gordijn@cs.vu.nl

Abstract

We present *control pattern*s: a framework for designing and analyzing inter-organizational control mechanisms, inspired by design patterns. A control pattern is a generic solution for some recurring control problem, applicable in a certain context. The patterns are based on internal control theory from the accounting and auditing fields, and on previous work on inter-organizational controls. The application of the patterns is supported by the *e³-control* methodology, which is based on the *e³-value* business modeling tool. The patterns are applied in a case study.

1 Introduction

In e-commerce, enterprises increasingly organize themselves as *value constellations*: a collection of enterprises that cooperate to satisfy a consumer need, each utilizing their own specific expertise, products, and services (Sarkar et al 1998; Timmers 1998; Tapscott 2000). One of the prerequisites for cooperation is that participants must *trust* each other not to behave *opportunistically*, i.e. default on their agreements. Opportunistic behavior also exists in a single enterprise, for instance when an employee commits fraud. Typically, such behavior is addressed by *organization controls*: measures to prevent, detect or correct opportunistic behavior (Starreveld et al 1994; Ronmey and Steinbart 2003). In a single enterprise, opportunistic behavior and the design of controls can be addressed by a single decision authority in a hierarchical way. This is impossible in a value constellation, because individual enterprises keep decision authority to themselves. Consequently, in value constellations different types of controls have to be developed, and designing controls becomes more like a negotiation process between equals, than following directives from an authority.

This paper presents a methodology for the analysis and design of *inter-organizational control mechanisms* for value constellations. The approach is based on *e³-control* (Kartseva et al 2005b), which is a value-based modeling approach for the analysis and design of inter-organizational controls. Please note that *e³-control* itself is based on the *e³-value* methodology for the analysis and design of value constellations (Gordijn and Akkermans 2001; 2003).

There are several reasons for using a *value*-based modeling approach. First, the value perspective is conceptually close to Transaction Cost Economics, which studies safeguards against opportunistic behavior in contract relationships (Williamson 1979). Because the *e³-value* and *e³-control* methods can handle value, they are capable of making a cost-benefit analysis of control mechanisms. This involves a risk assessment, which is typically part of auditing practice. 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 business models of the parties involved? These questions are not particularly relevant from an internal control perspective, but in a value constellation controls may affect the profitability of participants, or may even lead to new business opportunities. Examples of inter-organizational control mechanisms that are offered as separate commercial services, are Escrow services (Hu et al 2004), and the Letter of Credit Procedure (ICC 1993). Third, many controls themselves have inherent value-aspects. For instance, a Letter of Credit is a control document that may also be traded and re-sold. As far as we know, the value modeling approach has neither been applied in internal control, nor in inter-organizational control.

We organize existing domain knowledge about inter-organizational controls by so called *design patterns*. Design patterns have been invented in architecture (Alexander 1979) and have been successfully applied in computer science (Gamma et al 1995) to reuse design knowledge for the construction of information systems. In general, a design pattern is a description of a general and accepted *solution* for some recurring *problem*, which is applicable in a certain *context.* We call our patterns *control patterns*. Until now, *e³-control* has focused on constructs to model and conceptualize inter-organizational controls. Practitioners also need guidelines how to construct such models. The contribution of this paper is therefore to present accepted knowledge about inter-organizational controls for value constellations in a systematic way, by means of control patterns.

The domain knowledge in this paper comes in the first place from accounting and auditing theory (Starreveld et al 1994; Ronmey and Steinbart 2003, Schaad 2003). Well known principles are *segregation of duties* and *conflict of interest*. Other domain knowledge is taken from ongoing research on inter-organizational controls (e.g. Bons 1997; Lee et al 1998; Bons et al 2000; Weigand and de Moor 2003). Much of this work is based on the assumption that principles of internal accounting and auditing theory may be applicable to inter-organizational settings too, provided that the assumptions about the context in which such principles are applied, are made clear. A third source of domain knowledge is formed by a number of recent case studies in the fields of internet radio, renewable energy, international trade, and health care (Kartseva et al 2005a; Kartseva et al 2005b; Kartseva and Tan 2005).

The rest of the paper is structured as follows. In section 2, we provide some background on control patterns. In section 3, we study a number of particular control patterns from international trade. In section 4 we apply the patterns to a case study of an Escrow service.

2 Control Patterns

2.1 Patterns

Our proposal to use control patterns, is inspired by the use of design patterns in architecture (Alexander 1979), and especially in software development (Gamma et al 1995). A *design pattern* is a description of a general accepted solution for some recurrent problem, which is applicable in a certain context (Gamma et al 1995). Patterns are applied as follows: to solve a problem, a solution is selected, provided a number of constraints on the context hold. This selection process can be described by the following type of IF-THEN rules:

IF problem *P* AND context C, THEN solution *S*

To apply patterns in inter-organization control, we first have to define what the problems, solutions and context are in the case of controls:

Definition 1. A *control pattern* is a *generic* and *re-usable* control mechanism for a recurring control problem, selected on the basis of aspects of the context of application. The structure of a control pattern is the following:

1. *context*: a description of the value constellation to be controlled, modeled from an *ideal* perspective, meaning that no one behaves opportunistically.
2. *problem*: a statement of one or more the *control problems*, illustrated by scenarios that demonstrate a risk for opportunistic behavior, represented by a *sub-ideal value model* and a *process model* for the value constellation.
3. *solution*: a control mechanism, described by guidelines and illustrations of their implementation in process models and value models. A solution may have different variations, along with *forces*, which are conditions to select these variations.

The **context** should provide a background to the control-problem and its solutions. The context explains what is considered to be the norm. Controls should prevent, detect or correct any behavior not allowed by this norm. To describe this normative ideal behavior, we state the enterprises and what *economic value* they should exchange. A control problem exists if there is some deviation of the prescribed exchange of economic value, e.g. the wrong products are delivered, or no products are delivered at all. Therefore we model the context by *ideal value* models (see section 2.2).

The **problem** consists of one or more control problems. Following the accounting literature, a *control problem* is defined by one or more risks. Typically, a risk for opportunistic behavior is illustrated by one or more scenarios that display a violation of the ideal model. Fraudulous or undesired behavior can be represented by a *sub-ideal value model*. This is a value model that explicitly records potential violations, from a value perspective (see section 2.3). To understand a control problem, and to select appropriate solutions, we have to understand the inter-organizational business processes associated with a sub-ideal value model. To represent process models we use UML activity diagrams (see section 2.4). For each domain, we need a vocabulary of operational activities. Section 3.4 contains a vocabulary for inter-organizational settings.

The **solution** describes a control mechanism, which prevents, detects, or corrects the control problem. In case a solution fully solves the control problem, it can be specified by means of an ideal value model plus an activity diagram that shows how the control mechanism works. Given the context and problem descriptions, different ways of implementing a solution are possible. For design patterns, such alternatives are known as *variations*. The reasons that determine which variation of a solution must be implemented, is called a *force*. Forces are constraints on the context of application. In the case study variations prove to be useful: they can express differences, which are not crucial enough to warrant a separate pattern. In that case the selection process is described by the following IF-THEN rules:

IF problem *P* AND context *C* AND force *F*,
THEN solution *S*, with variation *V*

Control patterns can be used both for analyzing and designing processes. In analysis, they highlight risks, and identify possible solutions. In (re)-design, they explicitly guide the construction of a new organizational set up or business process. Below, we briefly review the notation for expressing control patterns.

2.2 Ideal value models: e³-value

The *e³-value* methodology provides modeling concepts, to represent which parties in a value constellation, exchange which objects of economic value with which other parties (Gordijn and Akkermans 2003). Figure 1 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 with 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 exchanges**. For value objects, some actor should be willing to pay, which is shown by a **value interface**. A value interface models the principle of *economic reciprocity:* only if you pay, you can obtain the goods and vice versa. A value interface consists of **value ports**, which 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 exchange of value objects. Exchanges may lead to other exchanges by the same actor, as in the case of the seller, or lead to a **boundary element**.

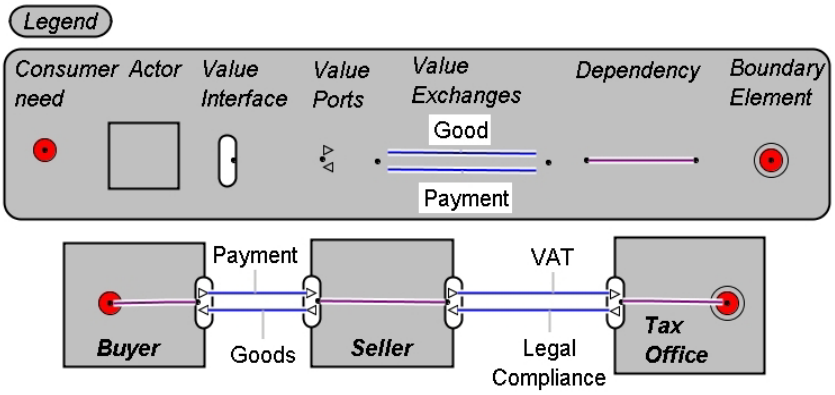

Figure 1: Example of an *e³-value* model of a purchase with payment of tax

2.3 Sub-ideal value models: e^3-control

In *e^3-value*, it is assumed that actors behave in an *ideal* way, meaning that all value exchanges occur as prescribed. This implies, amongst other things, that actors respect the principle of *economic reciprocity*. In reality, actors may not behave as stated in an *e^3-value* model; they can commit fraud or make unintentional errors. In *e^3-control*, which has been suggested as an extension of *e^3-value* (Kartseva et al. 2005b), these situations are modeled by **sub-ideal value exchanges**. These are graphically represented by a *dashed* line, which indicates the possibility of different risks, e.g. that actors will *not* pay for the goods, *not* obtain the goods, or obtain the *wrong* goods.

2.4 Process models

Control problems and control mechanisms often involve operational activities; not just value exchanges. Take for example the verification of an activity, or the transfer of an evidence document. The temporal order in which activities take place, forms a crucial part of controls (Chen 1992). So we need a graphical notation to represent the operational aspects of control problems and solutions. For this purpose we use UML-activity diagrams (Rumbaugh et al. 1999).

3 Inter-organizational Control Patterns

3.1 Internal control

Our inter-organizational control patterns are inspired by principles from the accounting and auditing fields (Starreveld 1994, Romney and Steinbart, Schaad 2003). Examples of internal control principles are *segregation of duties*, *decentralization* and *supervision and review* (Schaad 2003). Segregation of duties means that a critical transaction must be divided into at least two separate activities, performed by different actors. Decentralization means that one person cannot manage all activities; some activities must be delegated to subordinates. When activities are delegated, the supervision and review principle applies. This requires that all activities are reviewed post hoc by another person.

The internal control field provides rich and validated knowledge of risks and control mechanisms, but only in the scope of a hierarchical organization. It has been argued that these mechanisms can be approximated for inter-organizational settings too (Bons 1997). Extending the set of inter-organizational controls is one of the goals of our research. However, in this paper, we follow the practice in design patterns, to re-organize validated knowledge about solutions, rather than invent new solutions.

3.2 Trust

Trust plays a defining role in the analysis and design of inter-organizational controls. Parties in a business transaction may not trust each other, and be vulnerable to opportunistic behavior. That means that when parties invest in a transaction, they are uncertain whether the other party will perform their part of

the deal. In Transaction Cost Economics (TCE) this type of opportunistic behavior is referred to as 'ex-post' opportunism (Williamson 1979). It has been argued that such uncertainty can be reduced by supporting a business transaction with documents that provide a form of control, which is referred to as *documentary control*. Along a similar line, Bons (1997) argues that the exchange of evidence documents can reduce the risk of non-performance as it decreases the uncertainty of the parties involved.

3.3 Evidence Function and Preventative Function

We start from work on inter-organizational control mechanisms developed by Lee and Bons in various publications (Lee and Bons 1996; Bons 1997; Lee et al 1998). Their aim was to develop a set of re-usable trusted trade procedures for electronic data interchange (EDI). The control principles are validated with case studies from international trade, in particular the Letter of Credit procedure. In designing controls, Lee and Bons identify a *primary activity* performed by a *primary actor*, and a *counter activity* by a *counter actor*, that must be performed in return. The primary actor does not trust the counter actor; i.e. is expecting a risk. The risk is a situation in which, because of fraud or unintentional errors, the counter activity is executed inappropriately or not at all, while the primary activity has been executed.

Inter-organizational documentary controls have two objectives: a *preventative function* and an *evidence function*. The evidence function means that the party responsible for the performance of some activity should receive evidence of the activity being executed. This results in the following principle of Bons[1]:

> *General principle 1: If a primary activity is performed by Role1, Role2 should testify the completion thereof using some document, which should be received by Role1. If the party playing Role2 is not trusted by the party playing Role1, the primary activity should be executed after receiving the document. (Bons 1997; p .60)*

The preventative function means that before an actor executes an activity, for which some counter activity has to be performed by another actor, it should be certain about the performance of the counter activity.

> *General principle II: Before Role1 executes a primary activity, it should have witnessed the performance of the counter activity by some Role2, if the party playing Role1 does not trust the party responsible for Role2, unless it has received evidence that Role 2 has executed its tasks. (Bons 1997; p .61)*

If such witnessing is impossible and the reliability of the evidence is questionable, a trusted third party may be involved:

> *General principle III: If Role1 cannot witness the performance of a counter activity, another Role3 should testify the completion of Role2's activity, if the party playing Role2 is not trusted by the party playing*

---

1 To remain consistent with *e³-value* terminology, in the rest of the paper we will use the term 'actor' instead of Bons's phrase 'Role' or 'party playing Role'.

Role1. This document must be received by Role1 before the execution of its primary activity, and the party playing Role 3 should be trusted by the party playing Role 1. (Bons 1997; p .61)

In addition, for the preventative function two other principles are defined, which deal with situations when some activities are *delegated* to other agents.

Note that these control principles also incorporate internal control principles (see section 3.1). For example, a requirement that witnessing an activity is done by a party different from a party that executes the activity is an instance of the segregation of duty principle.

3.4 Vocabulary

The patterns are formulated in a general way, using variables for the actors and activities. We use the following vocabulary, roughly based on Bons' terminology (Figure 3). There are two actors: a **primary actor (A1)** and a **counter actor**.(**A2**) The primary actor must perform a **primary activity** (**PA**), by means of transferring a value object, called **primary object** (**PO**) to the counter actor. The counter actor must execute a **counter activity** (**CA**), by means of transferring a value object called **counter object** (**CO***)*, to the primary actor. The transfer of a value object is represented by an operational activity, called **execute,** performed by the actor with the outgoing value port.

The control activities that are considered in this paper are limited to witnessing, or **verification** as it is called in the remainder of this paper to remain consistent with accounting terminology, and the exchange of **evidence documents (Doc)**. A verification activity can only be performed by an actor who has access to the (results of the) activity that is verified. In case evidence is needed from a distance, another actor should testify. By definition, any activity that produces an evidence document, is called **testify** in this paper.

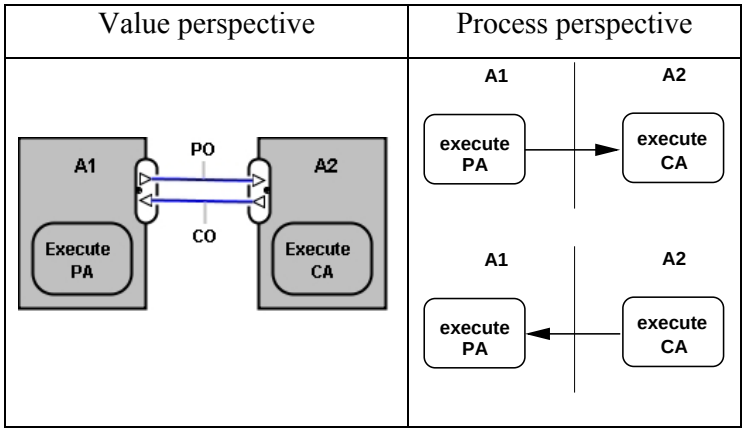

Figure 2: Value exchange between primary actor (A1) and counter actor (A2), with two possible process models

3.5 Patterns of Inter-organizational Control

Based on an overview of the accounting and auditing literature, we have developed a set of control patterns. Figure 3 contains the list of the current control patterns in our collection. Two patterns, `Pre-execution' and `Receipt', which are based on the principles of Bons described above, will be presented in more detail and applied in a case study.

Pattern	Description
pre-execution	require verification of counter activity before executing primary activity
receipt	require evidence of execution of primary activity from counter actor
post verification	require verification of counter activity after execution of counter activity
verification	verify correctness and completeness of the results of an activity, against given standards
commitment confirmation	require confirmation of commitment from counter actor before executing primary activity
commitment authorisation	require authorization to make commitment before accepting conformation of commitment
compensation	in case of sub-ideal exchanges, require compensation from the responsible actors

Figure 3. Collection of Patterns for Inter-Organizational Control

As defined in section 2, a pattern consists of a *name*, a *context*, a *control problem*, and a one or more *variations* of the *solution* (control mechanisms), with the corresponding *forces*, i.e. conditions to select a specific variation.

Pattern 'Pre-execution'

This pattern corresponds to the *preventative function* of documentary controls, and covers General Principle II and III of Bons (1997).

Context: A1 and A2 are about to engage in a transaction. According to contract, A1 must Execute PA and A2 must Execute CA.

Control problem: A1 does not trust A2, because of the risk that A2 may not properly execute CA, after having executed PA. This can be caused by fraud or unintentional error. For example, after having paid, a buyer may not receive the goods from the seller.

Control mechanism:

Force a. A1 is able to verify the execution of CA.

Variation a. Require that CA is executed before PA, and introduce a verification activity to check this. For example, the buyer inspects the goods, before payment.

Force b. A1 is not able to verify the execution of CA, or variation **a** is not acceptable to A2 (dead lock).

Variation b. Require an evidence document (Doc) of execution of CA, or of preparations for executing CA, before executing PA. This evidence can be obtained from a trusted third party (TTP). In that case, the value model must also be changed.

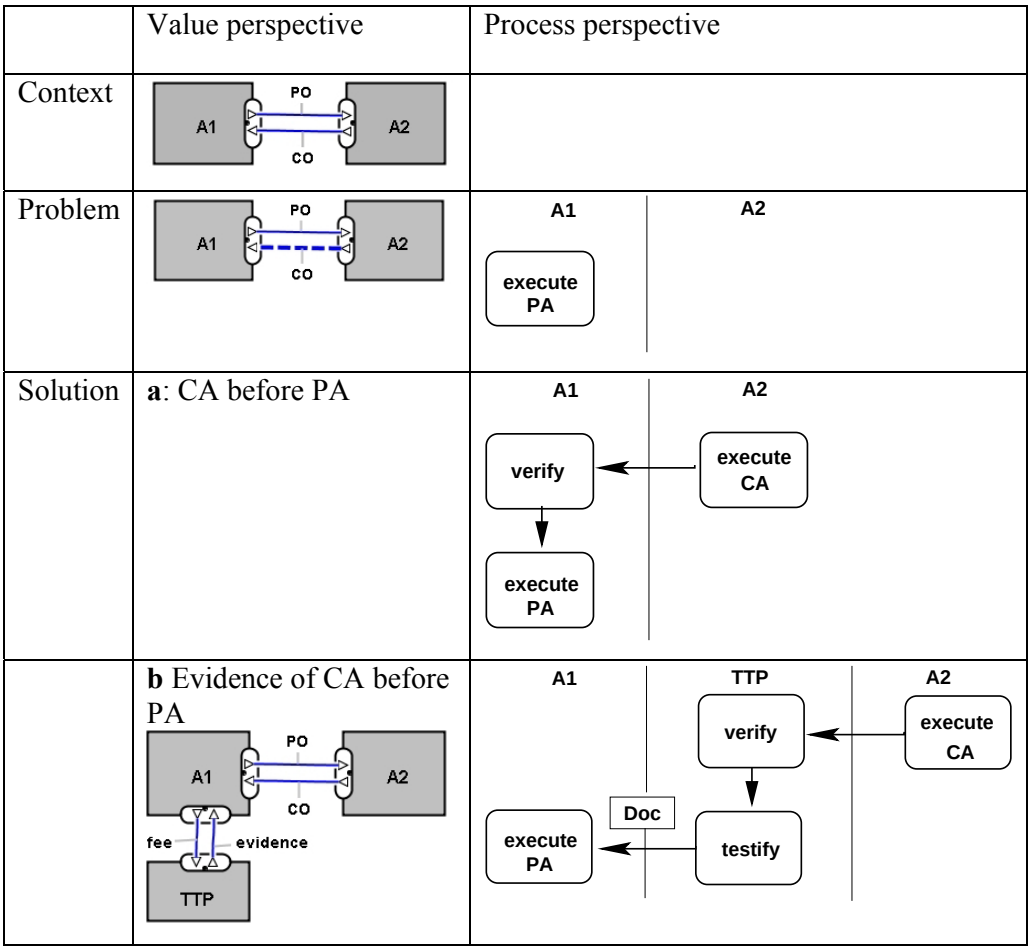

Figure 4: Pattern 'Pre-execution'

Pattern 'Receipt'

This pattern corresponds to the *evidence function* of documentary controls. It is based on General Principle I of Bons (1997).

Context: A1 and A2 are about to engage in a transaction. According to contract, A1 must Execute PA, and A2 must Execute CA.

Control problem: A1 does not trust A2. A1 runs the risk that if he cannot prove having executed PA, then A2 might behave opportunistically, by claiming that A1 did not execute PA and subsequently cancel CA, or claim reimbursement of CA.

Control mechanism: A1 requires an evidence document (Doc) from A2 that testifies execution of the primary activity.

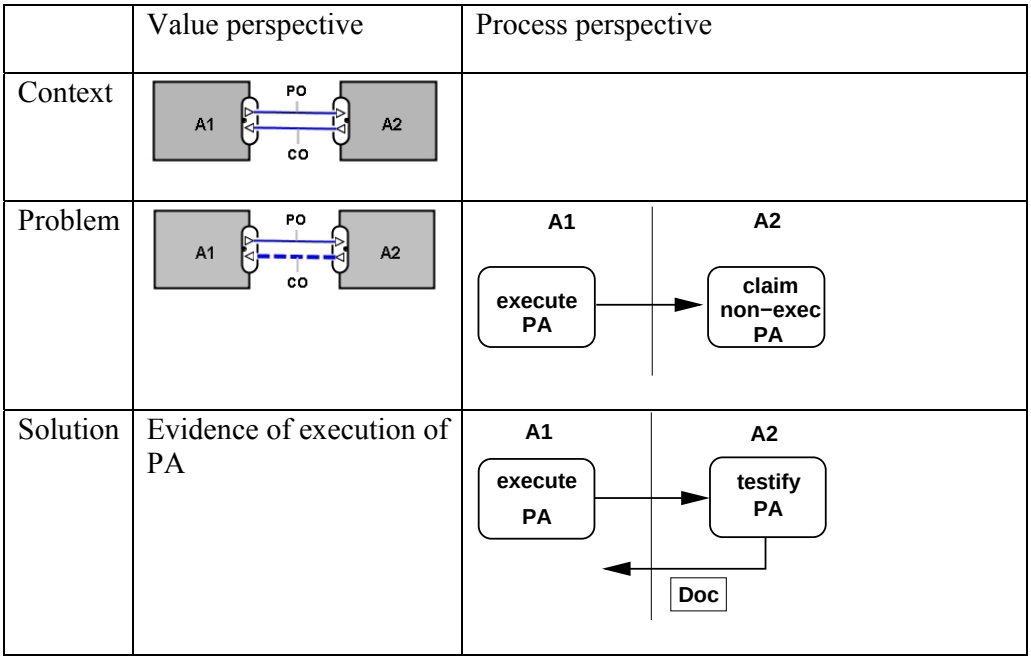

Figure 5: Pattern 'Receipt'

4 Case Study: International trade

An Escrow service is an inter-organizational control mechanism, to guarantee payment for a transaction. Escrow services can be delivered by a notary or a bank, but specific Escrow services exist for international trade, and electronic commerce. For more about the use of Escrow services on electronic markets, see Hu et al (2004).

In a typical scenario, buyer and seller have made some agreement about a transaction, and have also made an agreement about using and paying for the Escrow service. Now the buyer first transfers money to the Escrow provider. Once the payment is verified, the Escrow provider notifies the seller to ship the goods. The Escrow provider verifies whether the goods reach their destination, using tracking information provided by the carrier. After delivery, the buyer has a limited period of time to inspect the goods and accept the delivery. If the buyer accepts the goods, or waits beyond the deadline, the money is transferred to the seller. If the goods never arrive, or are returned by the buyer, the money is reimbursed to the buyer, with shipping costs deducted. Escrow providers may charge anywhere from 2% - 15% of the purchase price for their services.

Using an Escrow service reduces the risk for both seller and buyer. The buyer may distrust the seller's willingness to deliver the goods that were agreed on. Therefore the buyer can inspect the goods and return them. The seller may distrust the ability of the buyer to pay. Therefore the Escrow provider verifies the payment before

shipping. The Escrow provider also maintains a maximum period for the buyer to return the goods; after this period the seller will get the money anyway. In case of a conflict, the Escrow provider may mediate between the parties.

We are now going to construct a version of an Escrow service, using the patterns described in Section 3. Each step in the derivation starts from a description of a specific element of the context, which involves a control problem, caused by the distrust between the buyer and the seller. By applying a pattern, the situation is altered to avoid the control problem. Because not all issues can be addressed at the same time, patterns will have to be applied iteratively.

4.1 Step 1: General context description

A buyer and a seller have made an agreement about a transaction, which involves shipment of goods over a distance, in return for a sum of money. Buyer and seller do not know, and do not trust each other. The seller delegates shipping to a trusted carrier.

Figure 5 shows an ideal value model that represents the context. Buyer and seller exchange value objects Fee and Goods. The seller delegates Transportation carrier, in return for a Transportation Fee. The exchange of these value objects corresponds to the execution of the operational activities `pay goods', `ship goods', `pay transport' and `deliver goods'. At this stage buyer and seller have not agreed on a temporal order of these activities, which is why we do not show the activity diagram.

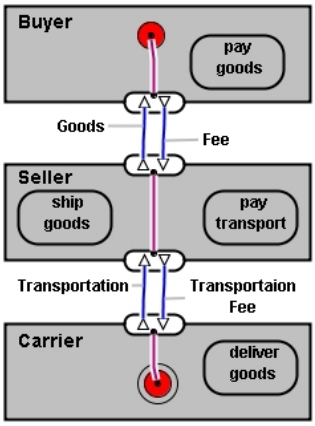

Figure 5: Initial value model with operational activities

4.2 Step 2: General control problem description

Now we look at possible risks in this ideal model. Every business exchange can have multiple risks. We consider only the types of risks that were discussed in the theory section. Each risk is modeled with a sub-ideal exchange in the value model in Figure 6. As is customary in *e*^{3}*-control*, the actor that is to blame for a sub-ideal exchange is labeled with a P (for penalty) and a number. First, we consider the risk that the buyer paid for the goods, but did not receive them (P1). This is a risk caused by opportunistic behavior of the seller. The other two risks are caused by opportunistic behavior of the buyer. The second risk is that the buyer will not pay

(P2). The third risk is that the buyer can claim that the wrong goods were delivered, claim reimbursement (P3).

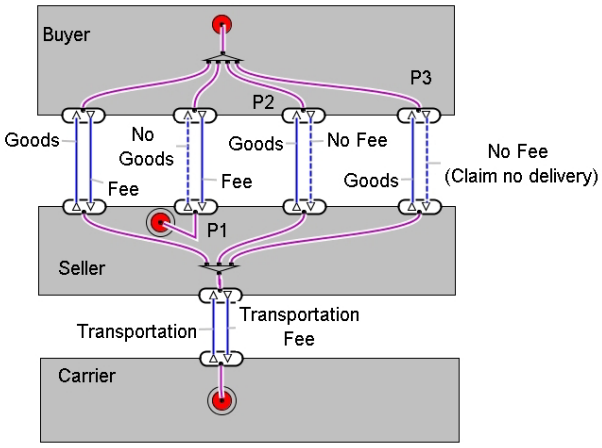

Figure 6: Sub-ideal value model, with four risks

4.3 Step 3. Control mechanism design

Risk P1: buyer does not trust seller about delivery

Control problem: After the buyer paid, the seller may not deliver the goods, or the wrong goods.

Trust relations: The buyer does not trust the seller or the carrier.

Pattern: Pre-execution

Assignment of variables: A1 = buyer, A2 = seller, PA = pay goods, CA = deliver goods

Control mechanism: The buyer requires (evidence of) delivery of the goods, before payment.

Forces: The buyer is able to verify the delivery. Apply variation a.

Application: In the process model (Figure 5) we add activity 'verify' for the buyer. Verification should be performed by an actor different from the seller. It cannot be assigned to the carrier, because the carrier is not trusted by the buyer. The value model remains unchanged, except that the operational activity 'verify' is added to the buyer.

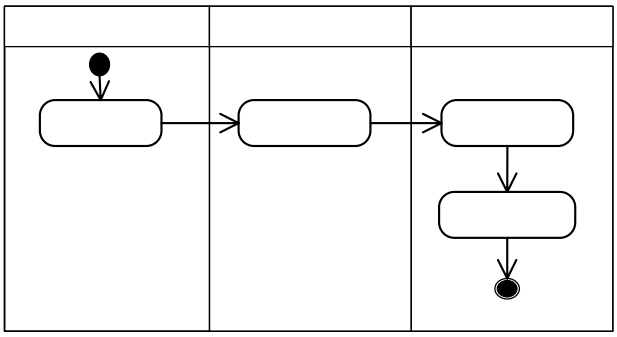

Figure 7: Process model after designing controls to prevent risk P1

Risk P2: seller does not trust buyer about payment

Control problem: After the seller delivered the goods, the buyer may not pay.

Trust relations: The seller does not trust the buyer; the seller trusts the carrier.

Pattern: Pre-execution

Assignment of variables: A1 = seller, A2 = buyer, PA = deliver goods, CA = pay goods.

Control mechanism: The seller requires (evidence of) payment, before shipping the goods.

Forces: The seller is able to verify payment. Apply variation a. This requires that 'deliver goods' is executed after 'pay goods'. That leads to a *deadlock* because the control mechanism against risk P1 prohibits this order. So other options need to be tried.

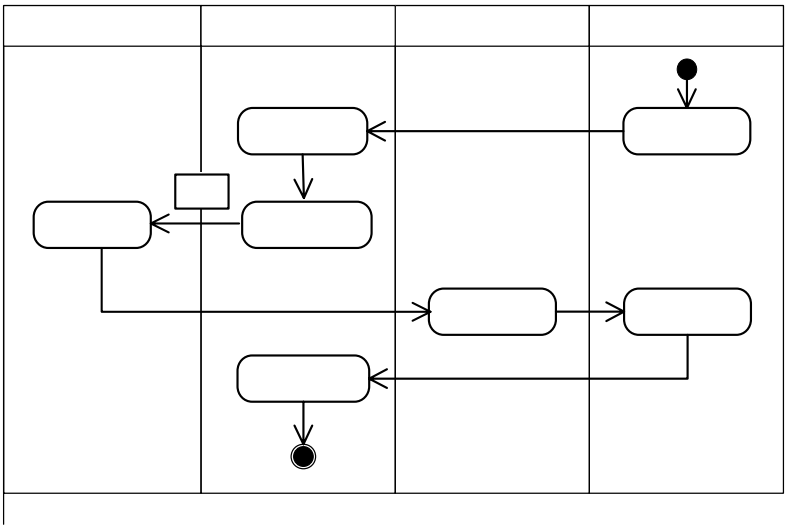

Figure 8: Process model after designing controls to prevent risk P2

We add a trusted third party: the Escrow. To avoid the deadlock we *split* the original 'pay goods' into two activities: 'down pay', from buyer to Escrow, and 'pay out' from Escrow to seller. Now we apply pattern pre-execution, variation b.

Assignment of variables: A1 = seller, A2 = buyer, TTP = Escrow, PA = deliver goods, CA = down pay.

Application: The Escrow provides evidence to the seller of the down payment. This mitigates risk P2 of the seller (Figure 8)

The split means that the buyer has delegated to the Escrow paying out the money. Payment still only occurs after the buyer has verified the delivery of the goods. This mitigates risk P1 of the buyer, as before.

In the value model an Escrow actor is added, who delivers the 'Escrow_service' in return for an 'Escrow_Fee' (see Figure 9). If both parties pay part of the fee,

Seller Escrow verify Doc ship goods testify

an additional value exchange would have to be added between Escrow and seller. The added operational activities are shown in white.

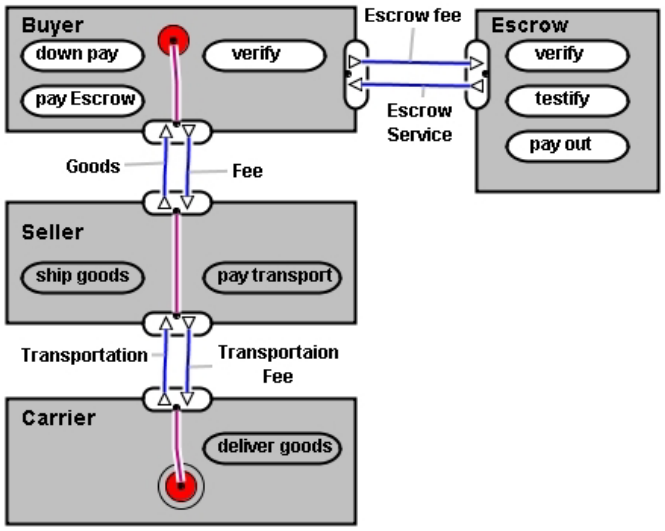

Figure 9: Value model after designing controls to prevent risk P2

Risk P3: seller does not trust buyer about receiving the goods

Control problem: After delivery, the buyer may claim that the goods were not delivered, and may delay pay out by failing to verify delivery

Trust relations: The seller does not trust the buyer. The seller trusts the carrier.

Pattern: Receipt

Assignment of variables: A1 = seller, A2 = buyer, PA = deliver goods, CA = 'verify' leading to 'pay out'.

Control mechanism: The seller requires an evidence document from the buyer testifying that the goods were delivered, immediately after delivery.

Application: Add a 'testify' activity from the buyer to the carrier (and therefore also to the seller), occurring immediately after delivery of the goods. There is a risk that the buyer may not provide this evidence document, because it goes against his own interests. After all, the buyer has not had access yet to the goods. Therefore, we can let the carrier make sure that a receipt is signed, before handing over the goods. In fact, this means that the activity 'deliver goods' is split up into two separate activities: 'arrival of goods', meaning that the goods arrive at the location of the buyer, and 'hand over goods', meaning that the goods are put into the custody of the buyer, followed by another application of the Pre-execution pattern (CA = testify, PA = hand over goods). Note that the buyer can testify arrival, but can only verify whether the right goods were sent after a longer inspection period (see below).

On the value level, the model remains the same as in Figure 9, except for adding 'testify arrival' to the buyer, and replacing 'deliver goods' by 'arrival goods' and 'hand over goods', for the carrier.

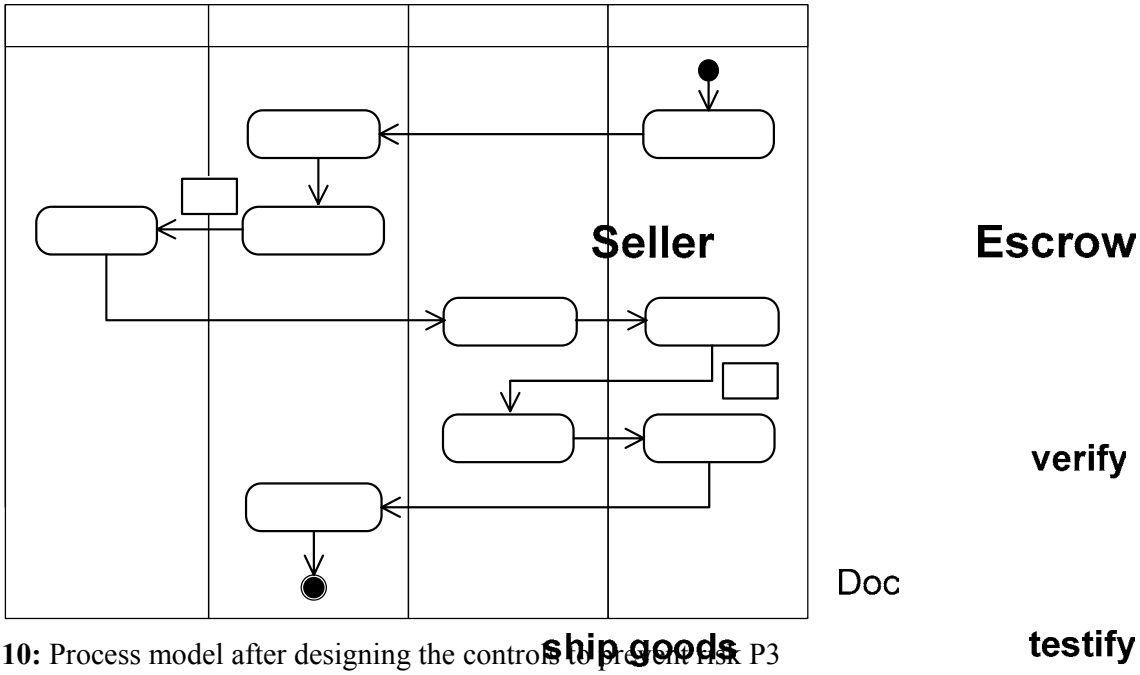

Figure 10: Process model after designing the controls to prevent risk P3

Further Risks

Often buyers have a right to return the goods, in case they are not satisfied. In that case, the Escrow must reimburse the payment. This could be modeled as another control pattern, to address the risk of the buyer that the wrong goods were delivered (similar to P1). But to protect the seller, the Escrow will then typically limit the duration of the inspection period during which goods may be returned. The application of the `Receipt' pattern above, creates evidence to mark the beginning of this period. If the period expires, the Escrow will execute 'pay out' anyway. After such extensions we arrive exactly at the Escrow service, as described at the beginning of the section. It shows that the design of an inter-organizational control mechanism is often a negotiation process.

Further control problems arise, when the strategic effects of time and money are taken into account. For example, in the current version, the buyer has to pay up front. In case the delivery takes a long time, this may be a risk for the buyer. For that reason, credit facilities may be added. In that case, the initial deposit (down pay) is financed by a credit from a bank. Adding a credit facility, with proper guarantees, would produce a form of the Letter of Credit procedure (Bons 1997; ICC 1993).

5 Conclusions

In this paper, we introduce a framework for applying control patterns. A control pattern is a generic and re-usable solution for a recurring control problem, selected on the basis of aspects of the context of application. The approach is inspired by the use of design patterns in software engineering. Design patterns must be based on well-established knowledge. Hence, we took the well-established internal control theory, and in particular the more inter-organisational interpretation of Bons thereof, as the theoretical basis for our control patterns.

The control patterns employ *value models* to describe the impact of a redesign on the structure of a value constellation. This value perspective is conceptually close to Transaction Cost Economics, which traditionally deals with measures against opportunistic behaviour. The value perspective enables a cost-benefit analysis of control mechanisms, including a limited form of risk analysis. Finally, in inter-organizational settings control mechanisms often appear as separate commercial services, altering the value constellation.

An interesting additional contribution of this paper is the way we combine the value activities of *e³-control*, with operational activities in regular UML activity diagrams.

A case study of an Escrow service illustrates how the patterns can be applied. An Escrow service is a way of overcoming the distrust between parties, who both require evidence of the other party's activity, before executing their own. From a control perspective, an Escrow service is a simplified Letter of Credit procedure, namely without a credit facility. This is in line with previous case study research.

Acknowledgements. The research of the first author is funded by the Post Master EDP Audit Education of the Vrije Universiteit Amsterdam. This research was partially funded by the BSIK Freeband project FRUX.

References

Alexander, C (1979): "The Timeless Way of Building", Oxford University Press, Oxford.

Bons, R.W.H (1997): "Designing Trustworthy Trade Procedures for Open Electronic Commerce", PhD Thesis, Erasmus University Rotterdam.

Bons, R.W.H., Dignum, F. Lee, R.M. and Tan,Y.-H. (2000): A formal analysis of auditing principles for electronic trade procedures. International Journal of Electronic Commerce, Vol. 5, No 1, pp. 57–82.

Chen, K.T. (1992) "Schematic Evaluation of Internal Accounting Control Systems", PhD Thesis, University of Texas at Austin, USA.

Gamma, E.R. Helm, R. Johnson, Vlissides, J. (1995): "Design Patterns", Addison-Wesley.

Gordijn, J., Akkermans, J.M. (2001): *e³-value*: Design and Evaluation of e-Business Models. IEEE Intelligent Systems, Special Issue on e-Business, Vol. 16, No. 4, pp. 11-17.

Gordijn, J., Akkermans, J.M. (2003): Value based requirements engineering: Exploring innovative e-commerce idea. Requirements Engineering Journal Vol. 8, No. 2. Springer Verlag, Berlin Heidelberg New York, pp. 114-134.

Hu, X., Lin, Zh., Whinston, A.B., Zhang, H. (2004) : Hope or Hype: On the Viability of Escrow Services as Trusted Third Parties in Online Auction Environments, Information Systems Research, Vol. 15, No. 3, pp. 236–249.

ICC: Uniform Customs and Practices for Documentary Credits 1993 (UCP500 1993), International Chamber of Commerce, ICC Publishing S.A., Paris.

Kartseva, V., Gordijn, J., Tan, Y.-H. (2005a): Designing Control Mechanisms for Value Webs: The Internet Radio Case Study, Proceedings of the 18th Bled Electronic Commerce Conference – eIntegration in Action (Bled'05), CD-ROM.

Kartseva, V., Gordijn, J., Tan, Y.-H. (2005b): Towards a Modelling Tool for Designing Control Mechanisms in Network Organisations, International Journal of Electronic Commerce, Vol 10, No 2, pp. 57-84.

Kartseva, V. and Tan, Y.-H. (2005): Designing Controls for a Marketplace of Health Care Services: a Case Study, Proceedings of the 12th Research Symposium on Emerging Electronic Markets (RSEEM 2005), CD-ROM.

Lee, R.M. and Bons, R.W.H. and Wagenaar, R.W. (1998): Designing trustworthy interorganizational trade procedures for open electronic commerce, International Journal of Electronic Commerce, Vol. 2, No. 3, pp. 61–83.

Osterwalder, A. and Pigneur, Y. (2002): An E-business model ontology for modeling E-business. In Proceedings of the 15th Bled Electronic Commerce Conference – eReality: Constructing the eEconomy (Bled'02), pp. 75–91.

Ronmey, M.B.; Steinbart, P.J. (2003): "Accounting Information Systems (9th edition)", Prentice Hall, New Jersey.

Rumbaugh, J., Jacobson, I. and Booch, G. (1999): "The Unified Modelling Language Reference Manual", Addison Wesley Longman, Reading, MA.

Sarkar, M.B., Butler, B. and Steinfield, C. (1998): Cybermediaries in electronic marketspace: Toward theory building. Journal of Business Research, Vol. 41, pp. 215–221.

Schaad, A. (2003): "A Framework for Organisational Control Principles", PhD Thesis, Department of Computer Science, University of York.

Starreveld, R.W., B. de Mare, and Joels, E. (1994): "Bestuurlijke Informatieverzorging", Deel 1, 4th edn., Samsom, Alphen aan den Rijn (in Dutch).

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

Timmers, P. (1998): Business models for electronic markets, Electronic Markets, Vol. 8, No. 2, pp. 3–8.

Weigand, H., and de Moor, A. (2003): Workflow analysis with communication norms. Data and Knowledge Engineering, Vol. 47, No. 3, pp. 349-369.

Williamson, O.E. (1979): Transaction Cost Economics: The governance of contractual relations, Journal of Law and Economics, Vol. 22, pp. 3-61..