Value-based Design of Collaboration Processes for e-Commerce

Pascal van Eck
Department of
Computer Science
University of Twente
P.O. Box 217
7500 AE Enschede
The Netherlands
vaneck@cs.utwente.nl

Jaap Gordijn
Department of
Computer Science
Vrije Universiteit
De Boelelaan 1081
1081 HV Amsterdam
The Netherlands
gordijn@cs.vu.nl

Roel Wieringa
Department of
Computer Science
University of Twente
P.O. Box 217
7500 AE Enschede
The Netherlands
roelw@cs.utwente.nl

Abstract

Designing cross-organizational e-business applications faces the problem that the collaborating businesses must align their commercial interests without any central decision-making authority. The design process must therefore yield a clear view of the commercial value of the collaboration for each economic actor, as well as a clear specification of the activities to be performed by each actor and a specification of information systems to be used by each actor. In this paper we present guidelines for designing the value network of the collaboration, which shows the commercial value of the collaboration for each participating actor. We then present guidelines for transforming the value network into process models, which show the feasibility of implementing the value network in the business processes of the actors. Our approach has been developed in different consultancy projects. We illustrate our approach with a consultancy project performed at a company that we will call the Amsterdam Times.

1. Introduction

Emerging standards such as ebXML [6] and web services [1, 4, 2, 12] offer notations in which we can specify how two or more actors can collaborate to produce joint results. The availability of implementations of these standards opens up a new field for requirements engineering: Cross-organizational requirements engineering. Traditionally, and also for the vast majority of systems currently built or acquired, system requirements originate from one organization. Whatever the internal organization structure, there is some form of decision-making hierarchy in which there is a single locus of authority that can authorize the requirements and can commence the acquisition of the system as specified by the requirements. In addition, the mission of such a system is, even if only indirectly, to contribute to the effectiveness or efficiency of one single organization.

By contrast, in cross-organizational requirements engineering, there is no single decision-making authority. There is a collection of two or more businesses that think that it is in the interest of each other to collaborate in some activities. In this paper, we restrict ourselves to a specific kind of collaboration: that of commercial transactions between actors (enterprises and end-consumers). This is often referred to as e-commerce.

To design a collaboration, each of the businesses must be convinced that the collaboration is in its own interest. This requires a clear specification of how the activities to be performed by each actor will contribute to the profitability of that actor. It also requires a clear specification of the cost items: the business processes to be performed and the information systems to be used in the collaboration. No business actor is likely to engage in entirely new business processes or acquire new information systems merely for this particular collaboration, so the process and system specifications must be sufficiently clear to estimate whether they can be implement using existing processes and systems.

To compound the problem, these decisions must often be made for business propositions that are initially very vague. For example, the e-commerce idea of the running example of this paper, the *Amsterdam Times*, is to offer its subscribers the service of accessing its news articles on line. Is this a profitable idea? Which business partners are needed to implement this idea? Which processes and systems do they need to implement it? Is there an implementation that is profitable for all actors involved? Part of the complexity is caused by the need to use technical knowledge to estimate the commercial feasibility of a business idea. This requires engineers and commercial managers to participate in a joint

decision-making process, and to overcome the communication problems that typically occur between these groups.

We approach this problem by abstracting it into several mutually overlapping viewpoints.

- The **value viewpoint** represents the collaboration from a commercial point of view by showing the value objects needed to satisfy the consumer need, by showing the value activities and value transfers that must take place among the actors to produce this consumer value, and by offering an estimate of the profitability of these activities.
- The **process viewpoint** represents the collaboration from an operational point of view by showing which activities must be performed to produce the value objects and transfer them among actors. The process viewpoint identifies operational activities to be performed by each actor.
- The **information systems viewpoint** shows the information systems needed to support the required business processes, and the data flows among the systems.
- The **infrastructure viewpoint** specifies the supporting infrastructure requirements by the information systems and the information exchanges among the actors.

These are all viewpoints on the same collaboration. One typically starts with developing the value viewpoint, because this is used to elaborate the initial e-commerce idea. Without this elaboration, we do not know for what purpose the business collaboration is to be implemented. But because these views are all views on the same collaboration, decisions made in each viewpoint may affect elements of other viewpoints. For example, it is possible that a decision made in the process viewpoint may affect the value viewpoint by, say, identifying a value activity that had previously gone unnoticed.

In this paper, we discuss the value and process viewpoints only. The value viewpoint was introduced in previous work [8, 7, 10] and we will present only a summary, which omits the profitability calculations. We focus in this paper on design guidelines for the value and process viewpoints.

Our research method is close to action research and thereby reflective learning. Starting from some initial ideas for e-business requirements engineering, four e-business consultancy projects were performed. After each project, the lessons learned where ploughed back into the requirements engineering method, and the resulting method was used in the next project. The ideas presented in this paper constitute the lessons learned from the Amsterdam Times project.

In section 2, we summarize the value viewpoint and give value design guidelines and in section 3 we give guidelines for elaborating this into a process specification. Section 4 then shows how process modelling leads to an update of the value viewpoint. Section 5 summarizes the paper and relates it to other work.

2. Collaboration Design Guidelines for the Economic Value Viewpoint

The economic value viewpoint provides the *raison d'être* for a business collaboration. It explains why collaboration between enterprises may work in the first place. It shows which actors are involved in the collaboration, what they transfer of *economic* value with each other, and what they request in return. In contrast, the process viewpoint is about how to put the economic value viewpoint into operation. As we will see, there are value design guidelines for a collaboration that are different from process design guidelines for a collaboration. Moreover, it is important to consider the design of a value viewpoint separately from the design of the business processes, because stakeholders that decide on business value issues (e.g. CxO's, or business analysts) differ from stakeholders that decide on process issues (workflow/business process designers).

To specify collaboration design choices from a business value perspective, we propose two related description techniques: (1) *value hierarchies* and (2) *value networks*. We introduce these techniques as well as guidelines how to use them in the upcoming sections. We kept these techniques deliberately simple to allow for easy understanding by stakeholders involved.

2.1. The Amsterdam Times Project

In this paper, we use a project about the *Amsterdam Times*, a company that publishes a newspaper as a running example. The *Amsterdam Times* has a subscriber base. The business idea is to offer subscribers services, such as accessing news articles on-line for free, along with surfing on the Internet and email. The commercial basis for the service at hand is the use of a *termination* fee to finance the on-line article service. In this context, *termination* is picking up the phone when someone calls. When a caller calls a callee, the telecommunication network sets up a connection path from caller to callee. When the callee picks up the phone, the termination point of this connection is realized and the telecommunications company can start collecting fees from the caller. If an actor is willing to cause termination of a large quantity of telephone calls, most telecommunication operators are willing to pay the actor for that. The price paid by the telecommunication operator per realized termination is called the *termination fee*. Because the *Amsterdam Times* has a large subscriber base, it is capable of generating a large number of terminations. Note that we are able to give a clear textual explanation of this

idea, which was only possible *after* making the value hierarchy and value network for this project. The stakeholders involved in exploring the e-commerce idea were not capable of articulating the idea this way initially.

2.2. Guidelines for Constructing a Value Hierarchy

A value hierarchy (see Figure 1 for an example) presents goods and/or services (collectively referred to as *value objects*), produced by collaborating enterprises, that satisfy a consumer need. Such a hierarchy consists of three basic elements:

- A **consumer need** to be satisfied; A human consumer need is a state of felt deprivation of some basic satisfaction [11]. For the case at hand, the need is to read a news article online.
- A **value object** that contributes to satisfaction of a consumer need. A value object is a service, a good, or money, which is of economic value for at least one of the actors involved [8].
- A **contributes-to relationship** between a consumer need and value objects, or between a value object and other value objects. The relationship states which value object(s) contribute to need satisfaction, or which value object(s) are required to create another value object. Variants of the *contributes-to* relationship may be used to model alternative, exclusive, or mandatory value objects that contribute to a need or another value object. Figure 1 shows AND-*contributes-to* relationships to represent that all objects are needed to create the upper object or consumer need. The reverse of the *contributes-to* relationship is the *consists-of* relationship. If A *contributes-to*, then B *consists-of* A.

The value hierarchy (Figure 1) says that to satisfy the need to read a news article, we need an online article and a telephone connection. Note that, in general, other connection mechanisms such as xDSL or cable can be used to create online access. However, in this specific case, we have chosen to only transport articles via a telephone based data communication connection. This is because the service is financed by termination fees, and these are only paid for telephone connections. Actually, it is exactly these decisions we want to represent by an object tree The on-line article can be offered if there is an article, a hosting service that stores the article, and IP access to the stored article. A telephone connection can be offered if we have a number of interconnections and a termination at the consumer.

Generally, it is important to state which kind of collaboration design choices we make in constructing a value hierarchy. A value hierarchy presents three collaboration design choices:

1. **Which consumer needs do exist?** By asking stakeholders to formulate a consumer need they plan to satisfy, we increase the chance that products and services are really wanted by consumers. It is our experience that many stakeholders have products or services in mind they want themselves, rather than those wanted by their customer. A similar approach is also suggested by Tapscott et al. [19].

   We assume that the consumer need is (at least vaguely) known. The value object hierarchy only states the need. To our experience, finding such a need is a highly creative marketing oriented process which is difficult to capture in guidelines. Consequently, finding such a need is outside the scope of our work, in contrast to refining and better understanding a given consumer need. Actually, this is one of the main purposes of modelling collaboration from a business value perspective: a better understanding of a given consumer need and how it can be fulfilled.

2. **Which (alternative) value objects satisfy a need,** and **which (alternative) value objects contribute to creation of another value object?**

   Value objects are goods or services that can be produced by an enterprise, or by a collaboration of enterprises. An important upcoming design choice is *who* is going to produce and consume a value object. Consequently, a first step is to identify those objects that can be produced and consumed independently from other objects by an actor.

3. **What is the scope of the collaboration: which value objects to include and which not?** In a value object hierarchy, leaves indicate the scope of the collaboration under study; we call such objects *contextual* value objects. The value objects which are leaves are assumed to be available already, and not part of the collaboration design study. As a consequence, value objects needed to produce the leaf objects are also not in the scope of collaboration design.

We use a number of guidelines to make these design choices. In this paper we indicate design guidelines with a tick ($\surd$) symbol.

$\surd$ **Use well-defined criteria for value object identification.** If a consumer need has been textually stated, value object(s) must be found. Value objects must satisfy each of these three criteria:

(a) obtaining a value object should contribute to satisfaction of a consumer need, or a value object should be usable to produce another value object and should add economic value to it;

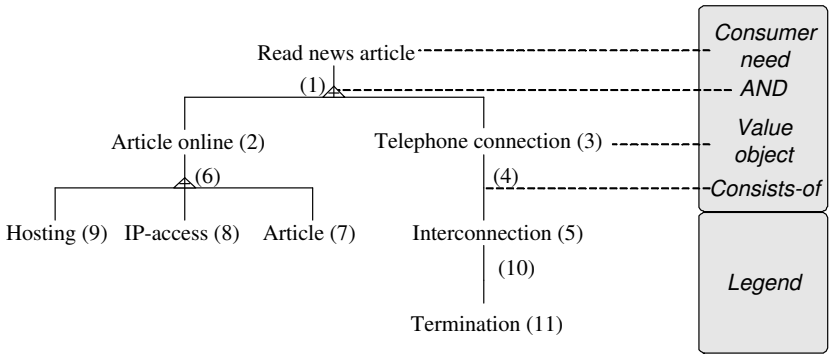


Figure 1. Value object hierarchy showing the objects satisfying a consumer need.

(b) a value object should be of *economic* value to someone;

(c) If the value object is a good, it should be possible to possess or rent it; if it is a valuable right, it should be possible to grant it to someone; and if it is a service, the outcome of performing the service delivery activity should be of value for someone.

We represent the satisfaction of a consumer need by a value object by relating the need and object by a *contributes-to* relationship.

√ **Find fine-grained value object by deconstructing coarse-grained objects.** Value objects can be split up into smaller objects that still satisfy the above criteria. Finding such smaller objects is of interest, because these objects might be produced by different enterprises, thus resulting in different collaborations. Consequently, to find smaller objects, we ask ourselves the question whether the candidate smaller objects can be supplied and/or consumed by different actors.

There are two directions we take to split up value objects (Figure 2). First we can split up a value object c that contributes to object a into two or more smaller ones (c_1 and c_2), but the smaller objects still directly contribute to object a. We call this a horizontal split up. In the project at hand, this was done for *IP access* and *Hosting*. Originally, these objects were one object called *Internet Service Provisioning*. Second, we may decide to split a value object c into two or more smaller objects (c_1 and c_2), but now the smaller objects directly contribute to object c, and indirectly to object a. This is called a vertical split up. Here, an object c still can be delivered by an actor, who obtains objects c_1 and c_2 from others, whereas in case of a horizontal split up, only object c_1 and c_2 can be provided.

Splitting up a value object into two or more smaller ones results in a new *contributes-to* relation between the value objects and its smaller parts. As we will see later on, the *contributes-to* relationship indicates an activity, to be performed by an actor, that constructs the smaller objects into the original object again.

√ **Stop with value object construction if the object under consideration falls outside the scope of analysis.**

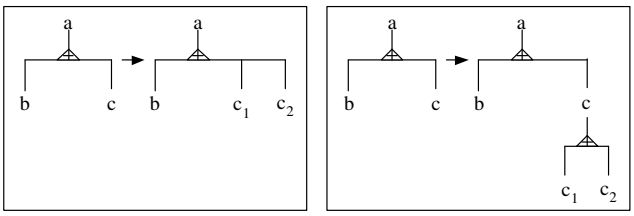
A value object needs not to be deconstructed further if we reasonably can assume that at least one actor can produce that object, and/or we not interested to analyze which other value objects are needed to produce the value object under consideration. Alternatively, it is possible that a given value object can not be deconstructed because no finer-grained objects can be found that comply with formerly mentioned criteria for value object identification.

2.3. Guidelines for Constructing a Value Network

A value network is a graph that represents a number of collaborating actors that create, distribute and consume value objects. Value networks add the following notions to those used in value hierarchies (Figure 3):

- An **actor** is perceived by his/her environment as an economically independent (and often also legal) entity. An **elementary** actor corresponds to one legal entity. Figure 3 shows a number of elementary actors: readers, a news paper called *Amsterdam Times*, and a local loop telecommunication operator *Last Mile*.

  In contrast, a **composite** actor contains two or more other actors, and consequently consists of more than one legal entity. For example, *Telecommunication consortium* is a composition of a web-hosting company (*Hoster*) and a telecommunication enterprise (*Datarunner*). They have a bundled offering to *Amsterdam Times* as is indicated by the value interface (see below) of *Telecommunication consortium*. In this specific case, combined *hosting* and *IP access* can be offered for lower fees than separately, because *Hoster* and *Data Runner* are located at the same



(a) horizontal split up (b) vertical split up

Figure 2. Horizontal and vertical split-up of value objects.

physical site and thus expenses for wide-area data connections between each other are saved.

- A **value port** is a mechanism by which an actor can provide or request value objects to or from other actors. The nature of this mechanism is left unspecified in the value viewpoint.
- A **value interface** is a set of ports of an actor that jointly behave atomically. This means that objects should be exchanged via *all* ports in an interface, or via none at all. The interface must contain at least one incoming and one outgoing port, and its incoming ports must balance its outgoing ports. This models economic reciprocity. For example, an interface may contain an in-going port for a payment and an out-going port for a good or service to be provided.

  The case at hand shows a number of value interfaces for the actors involved. The value interface of *Telecommunication consortium* shows what the partnership is offering to the outside world (access + hosting + termination fee), as well as the requested compensation for that offering.
- A **value transfer** is used to connect an outgoing value port of an actor to an incoming value port of another actor.
- A **value transaction** is a set of two or more reciprocal value transfers that jointly are atomic. It represents one or more potential trades of value object instances between value ports. Value transactions are the primary means of coordinating different business actors.
- A **dependency path**: If an actor exchanges objects of value via one of its value interfaces, that same actor may need to exchange value objects via one of its other value interface. For example, in a simple trading situation, the actor buys one object via one interface and resells the object via another interface. These internal relationships between interfaces of an actor are shown by a dependency path. A dependency path represents internal coordination within one actor. It is the counterpart of a value transaction, which represents an external coordination of several actors.

  For example *Amsterdam Times* has a dependency path modelling that the exchange of an article online must be coordinated externally with the exchanges with the telecommunication consortium and internally with article production. So the transfer of an article online to a reader requires coordination with hosting and access services, which are needed to deal with the technical aspects of article delivery. And it must be coordinated internally with the publishing activity to provide content. In fact, the dependency path in the *Amsterdam Times* shows that for N deliveries of an online article only one instance of the article publishing activity is needed.

  Note that the internal coordination represented by a a dependency path may be realized by several different business processes. This is a design choice to be made in the process viewpoint. If we are not interested in the internal coordination needed to participate in a value exchange, a dependency terminator ('T') is shown.
- **A value activity**: Actors perform activities to generate profit or utility. We separate the performer and the activity to facilitate the discussion of 'who does ' what. Note that whereas Porter's [16] *value activity* may be a cost center only, our interpretation of *value activity* requires that at least one actor expects to make profit by performing the activity.

As was the case for the value hierarchy, the value network shows a number of collaboration design choices:

1. **Who offers/requests which value object to/from its environment?** Each value object taken from the value hierarchy, potentially can be offered by different actors. By assigning a value object to an outgoing or ingoing port of an actor, we decide which actor offers or requests such an object.
2. Closely related with the previous question is the question **What are the value activities?** An actor offering value objects must perform activities to obtain these objects, for instance by manufacturing objects or by trading these. Since these activities generate profit it is important to decide on the performing actor.

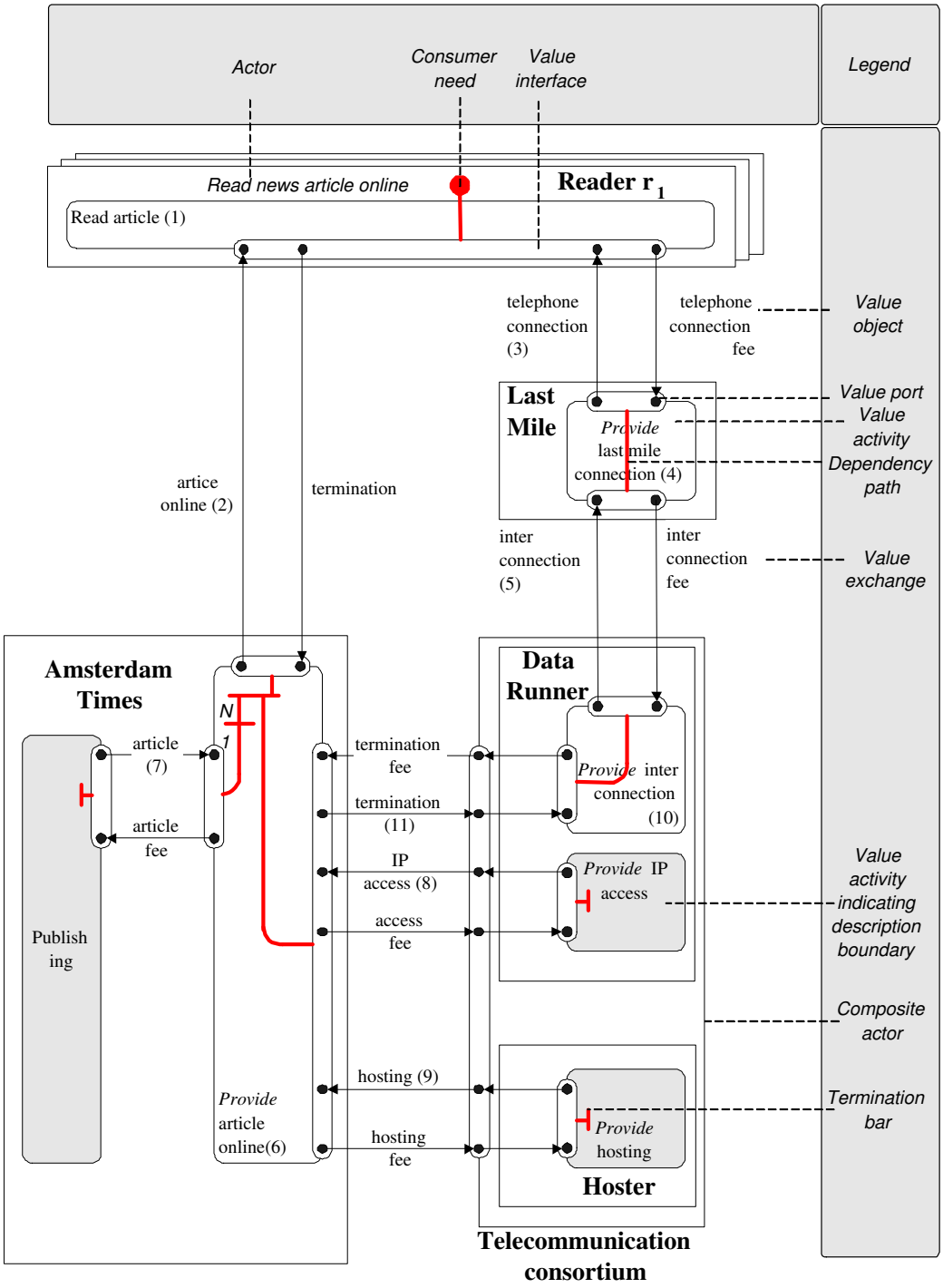


Figure 3. Value network of the Amsterdam Times business idea.

3. **What are the economic reciprocal value objects?** A value hierarchy only states which objects satisfy a need, not how someone who is offering such an object is compensated for that. We call such objects *economic reciprocal* objects. Economic reciprocity is shown by the value interface construct.
4. **Is there (un)bundling of value objects?** Apart from economic reciprocity, the value interface may also show bundling decisions. For instance, an actor may offer two (different) value objects as one offering to its environment. An actor may do so because he believes that he will generate more revenue by selling objects as a package deal, rather than selling them separately.
5. **Which partnerships exist?** To offer a specific coarse-grained value object, actors may decide to bundle more fine-grained objects. The important design decision is here that from a customer perspective, the coarse-grained object is offered by one (virtual) enterprise; and that the companies the virtual enterprise consists of are invisible for the customer.

The following guidelines may contribute to answering the above design questions:

√ **The *consists-of/contributes-to* relationships between value objects in the hierarchy indicate value activities.** A value object with mandatory *consists-of* relations to other value objects indicates a potential value activity. Such a value activity should produce value objects, using the value objects related by *consists-of* as inputs.

For example, the value object hierarchy shows that an article online (2) *consists-of* hosting (9), IP-access (8) and an article (7). Following our guideline, the value network should show a value activity that aggregates the inputs (objects 7, 8, and 9) into the output (object 2). This is the activity "Provide article online" (6).

√ **Leaves in the value object hierarchy indicate contextual value activities.** We have stated before that the leaves in the value object hierarchy correspond to contextual value objects: we are neither interested in how these objects are created and consumed, nor in profitable ways to do so. We assume that they are available. To be able to draw value networks, and more specifically value transfers, we need however *contextual value activities* (and actors) that produce the contextual value objects. So, contextual value objects may result in contextual value activities. In some cases, a value activity may produce both contextual and non-contextual value objects. Such an activity is then considered as a normal (thus non-contextual) value activity.

As an example of this guideline, the contextual value object *article* (7) requires in a contextual activity called *publishing*. For the case at hand, articles are already profitably published. Since there is already an existing process of the *Amsterdam Times* to produce a regular, paper-based, newspaper, the publishing activity is out of scope for our analysis. Contextual value activities are shown in grey in the value network. Since we are not interested in the internal coordination of contextual value activities, any dependency path in them terminates.

√ **Each value object is assigned to at least two ports of different actors or value activities.** A value object is assigned to at least one out-going port of an actor who offers the object and to at least one in-going port of an actor who requests the object. Moreover, these ports, are related by a value exchange.

√ **Value objects offered and value objects requested by a value activity, which are related by a *consists-of/contributes-to* relationship are each assigned to a separate value interface.** This follows from the logic of value activities and interfaces.

√ **Each offered/requested value object has a reciprocal requested/offered value object.** Value interfaces model economic rational behaviour: 'one good turn deserves another'. So, an interface contains at least one requested (in-going) and one offered (out-going) value object. For each value object, one asks what the reciprocal value object is.

For example, the value object *telephone connection* (3) offered by the value activity *Provide Last Mile connection* supposes a fee to be paid.

√ **If value objects are related in the hierarchy, they are related in the value network by means of dependency paths.** In a value network, we show the *consists-of/contributes-to* relationship stated in the value object hierarchy by a dependency path. This allows us to do profitability assessments: if we know the number of consumer needs per timeframe, we can calculate the total amount of objects leaving and entering an actor for that timeframe.

√ **Bundle objects if it is likely that they generate more profit in combination than sold separately.** The value interface construct can be used to express the notion of bundling, by showing two or more offered value objects into one value interface (or two or more requested value objects). We have encountered the following reasons to bundle objects into a value interface.

- First, an actor may believe that by selling two objects as one package, he will create more revenue than selling both objects separately (known

as mixed bundling [3]. We then bundle offered, out-going objects.

- Second, an actor may only assign value to having two objects in combination rather than having them separately. For example, consider the reader in Figure 3. Here, we bundle *article online* and *telephone connection* to express that these objects are only of value for the reader in combination.

√ **Create a partnership if enterprises together can offer more competitive conditions than on their own.** Closely related to bundling is the notion of partnership. Use a partnership if two or more actors decide to bundle their offered or requested objects to their environment. The already discussed telecommunication consortium is an example of such an actor.

3. Collaboration Design Guidelines for the Process Viewpoint

While the value viewpoint explains why a collaboration between enterprises may work from a commercial point of view, the process viewpoint serves as a feasibility study and completeness check of the value viewpoint. Developing the process viewpoint is the first step in the operationalization of the value network, as it identifies key processes that have to be performed by each actor in a value network. Moreover, identification of processes is a first step in the design of information systems that support each actor in performing its role in the value network. Second, as we will see later on, identification of processes may facilitate evaluation of the value network. Often, one or more of the processes found in this phase reveal value objects that need to be incorporated in the value network and that give rise to design questions in the value viewpoint as presented in the previous section.

3.1. Process Viewpoint Principles

The process viewpoint focusses on the internal processes that each actor has to execute to enable the value transactions it is involved in. The process viewpoint is developed from the value viewpoint in a systematic way, based on the following principles. First, we interpret each value transaction as the provisioning of a *service* by one actor, the service provider, to one of the other actors, the service consumer. Thus, in this interpretation, value transactions are asymmetric: one can distinguish a provider and a consumer. The actor that initially causes the value object to be transferred in the transaction is the service provider, the actor that is willing to pay a monetary fee for this is the service consumer. We assume that it is always possible to interpret one actor as consumer and the other as a provider, even in the case a value object is paid for by a value object that is not a monetary fee. Table 1 presents the services in the Amsterdam Times example.

Second, we develop a service delivery process for each transaction, identifying all process steps to be carried out by the service provider as well as the consumer to deliver the service. Several diagramming techniques can be employed to visualize the temporal relations between these process steps, such as simple flowcharts, Petri Nets or UML activity diagrams. (In this paper, we use UML activity diagrams. We draw all activity diagrams horizontally to preserve space.)

3.2. Guidelines for Constructing the Process Viewpoint

Figures 4 and 5 show activity diagrams for transaction (2) 'Article online' (depicted in Figure 3), the transaction in which the reader and Amsterdam Times exchange online news articles and termination. In Figure 4, the process steps of both a reader and Amsterdam Times are depicted that are needed to establish a service delivery relationship. In the case of Amsterdam Times, readers have to use an Internet dialup account with the telecommunication consortium to provide terminations. Thus, establishing a service delivery relation involves setting up this account. Specifically, to establish a service delivery relation, Amsterdam Times has distributed admission forms to its readers which have to be mailed back to Amsterdam Times to subscribe. (As a written signature is needed for legal reasons, only subscription by mail form is possible.) Amsterdam Times processes all forms, e.g. by checking whether the sender is indeed subscribed to Amsterdam Times. After that, an account for the reader must be set up with the telecommunication consortium. Amsterdam Times gets the account information (username and password) and forwards this information, together with a welcome letter and instructions, to the reader. The reader then must configure his or her computer so that it will use the newly created account for internet access to obtain online articles. The reader either succeeds in configuring his or her computer, or fails. In the latter case, a helpdesk is available to provide help.

After the reader has succeeded in configuring his or her computer (as depicted in Figures 4), the service delivery relation is established and actual service delivery can commence (Figure 5). In the case of Amsterdam Times, service delivery consists of two process steps: publishing an article by Amsterdam Times and downloading it by the reader. One reader can download many articles, as indicated by the arrow back to 'Download article'. In case of problems, the helpdesk is available. Moreover, Amsterdam Times may decide to monitor quality of service using a standard application monitoring facility. In this case, an extra process step

Transaction	Value object	Consumer	Provider
Article online (2) - termination	Article	Reader	Amsterdam Times
Telephone connection (3) - fee	Connection	Reader	Last Mile
Interconnection (5) - fee	Connection	Last Mile	Data Runner
Article (7) - fee	Article	Amsterdam Times	Publishing
IP Access (8) - fee	IP access	Amsterdam Times	Data Runner
Hosting (9) - fee	Webspace	Amsterdam Times	Hoster
Termination (11) - fee	Termination	Data Runner	Amsterdam Times

Table 1. Service provisioning interpretation of value transactions. (Numbers refer to numbers in the value object hierarchy depicted in Figure 1.)

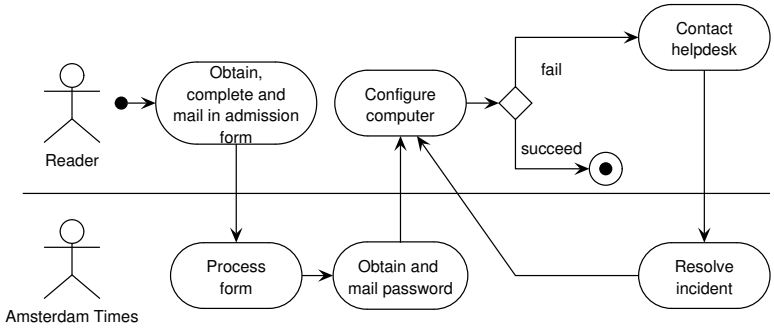


Figure 4. Activity diagram for establishing the relationship required for performing the transaction 'article online'.

will be included in the activity diagram, which runs concurrently with the 'publish article' process step.

A complete process viewpoint contains activity diagrams for all transactions in Table 1. Moreover, these activity diagrams contain more detail than is provided in the diagrams presented in this paper. (For instance, in this paper all process steps for ending a service delivery relationship have been left out.) Nevertheless, this section shows how the process viewpoint elaborates upon the value network.

In the construction of the process view, we have to make a number of design choices. These choices amount to the following questions:

1. **Which process steps have to be executed by the service provider and/or the service consumer to establish the service delivery relationship?** We assume that each service is delivered, most often more than once, in the context of an enduring relationship between the provider and the customer. Therefore, before a service can be delivered for the first time, the associated relationship has to be established, for instance by sending in a subscription form and/or paying the first term of a membership fee.
2. **Which process steps have to be executed by the service provider and/or the service consumer to actually deliver/consume the service, each time it is used?** The delivery of a service needs process steps to be performed by the provider, but also by the consumer. It is important to consider both. Take as an example a grocery delivery service. First and foremost, the provider has to carry out a number of process steps: picking the order in its warehouse, dispatching a truck, etc. The service consumer, in this case a household, also has to carry out a process step: receiving the groceries once they are delivered. If this process step fails, for instance because no one is at home, service delivery as a whole has failed. Therefore, when designing a service, process steps of both the service provider and consumer have to be taken into account to evaluate feasibility.
3. **Which process steps have to be executed by the service provider and/or the service consumer to end the service delivery relationship?** Sooner or later, a service consumer starts to consider whether he or she is still interested in the service offered. The service provider may want to perform process steps to keep the customer for later business (either for the same ser-

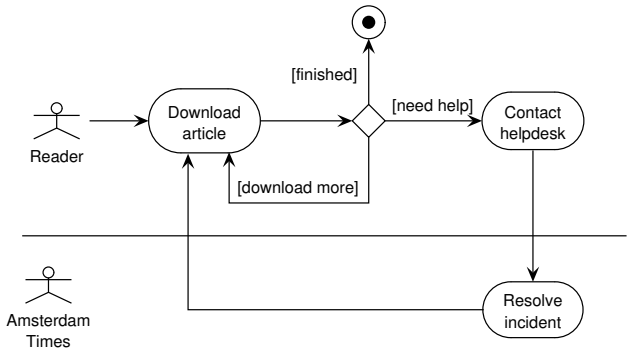


Figure 5. Activity diagram for transaction 'article online'.

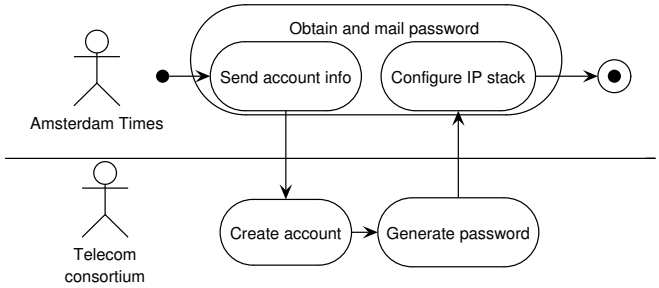


Figure 6. Activity diagram for obtaining password.

vice or for a substitute). If the service is actually terminated, process steps have to be executed for instance to settle final payments.

4. **What is the quality of service (QoS) level that the service provider offers to the consumer, and what process steps have to be executed to reach and maintain this level?** Quality of service should be interpreted here in a broad sense, ranging from technical parameters such as availability and response time of the servers that provide an electronic service, to for instance the aesthetic quality of tangibles such as printed instruction booklets. Process steps needed to reach and maintain the level offered generally can be divided in three categories: monitoring, prevention and correction. Monitoring in electronic service delivery is often provided by standard components (web application monitors). To monitor for instance customer perception of quality of service, surveys can be conducted. Preventive actions are for instance installing extra hardware whenever usage levels reach a specific threshold. Corrective actions can be: incident management as performed by a helpdesk or debugging programs [17]. The activity diagram in Figure 4 shows that in the case of Amsterdam Times, it has been decided to provide a helpdesk. This choice needs to be further refined by answering questions like whether a toll-free helpdesk number will be provided and what maximum waiting time is still acceptable.

To answer these questions, we use the following guidelines:

- √ **Service delivery relationship setup requires exchange of identification and authorization data.** In a value network, a number of actors call upon each other to deliver services that have a certain economic value. In many cases, a service provider will only be willing to start delivery of a service if it is reasonably sure that it will be rewarded. One way to provide such assurance is via a third actor, who identifies the consumer as being genuine and trustworthy and provides identification data.
- √ **Service provider process steps may themselves be realized by consuming services offered by a third actor in the value network.** A value network explains how a number of actors cooperate to jointly deliver a service for an end user, by each performing those process steps that it is best suited to carry out. In the service delivery perspective taken in the process viewpoint, this amounts to actors consuming services provided by other actors. In the case of Amsterdam Times,

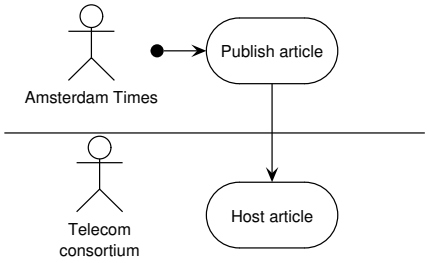


Figure 7. Activity diagram for service delivery of transaction 'hosting'.

process step 'Obtain and mail password' in Figure 4 can be decomposed as indicated in Figure 6. In Figure 6, Amsterdam Times plays the role of service consumer, while the telecommunication consortium plays the role of service provider. The activity diagram in Figure 6 establishes the service delivery relationship for transaction 'termination (11)' in Table 1. Thus, Amsterdam Times consumes one service to provide another one. Note that this refinement of "Obtain and mail password" requires an update of figure 4. Process step 'publish article' in Figure 4 is also realised by consuming a service provided by the telecommunication consortium, as depicted in Figure 7. This is the service delivery associated with transaction 'Hosting (9)' in Table 1.

√ **Process steps may be needed to ensure atomicity of a value transaction.** As explained in Section 2, a value transaction is atomic. In some cases, process steps may be needed to manage concurrent occurences of transactions and/or to roll back or compensate process steps that are already performed in case a transaction is aborted.

4. The value hierarchy & value network revisited

The activity diagrams presented in the previous section introduce a process step, *resolve incident* (see Figures 4 and 5), that raised discussion amongst the stakeholders involved. The *Amsterdam Times*, the intended performing actor for this process step, does not have sufficient resources and/or skills to execute this process step. Consequently, a new actor has to be found, who executes the *resolve incident* process step, but now as a *value activity* with the aim to generate profit.

To generalize, we see that the operationalization of a value network by means of a process model may reveal new value activities (and value objects), which causes us to revisit the design decision of the value viewpoint. We have seen this also in a previous case study on Internet contact advertising [9]. In short, making a process model contributes to the soundness and comprehensiveness of a value network

Figures 8 and 9 show the revised value object hierarchy and value network if we consider the *resolve incident* process step as a value activity. The value object hierarchy shows that an article online consists now also of zero or more solved incidents (the parentheses surrounding *solved incident* indicate optionality). Incident solving has been outsourced to an external party (*The Helper*) and *Amsterdam Times* pays a fee for this. The OR-dependency construct presents that transfers for a solved incident are optional.

5. Discussion and Conclusions

In this paper, we have presented guidelines for the design of collaboration processes for e-commerce for two viewpoints: the value viewpoint and the process viewpoint. The value viewpoint identifies actors, value activities and exchanges needed to satisfy a top-level consumer need. The process viewpoint identifies all processes needed to perform these activities and exchanges. The viewpoints together integrate commercial and technical views of an e-commerce idea, and we introduced a number of simple description techniques that can be understood by all stakeholders and are yet precise enough to analyze commercial and technical feasibility of the idea. The proposed process can be viewed as a kind of commercial-technical co-design of e-business systems. The approach has been validated in a number of consultancy projects in the field of internet service provisioning, news, ads, energy, music and banking.

The guidelines for the transition from the value viewpoint to the process viewpoint presented in this paper can be compared to a number of other approaches, such as found in traditional business process modeling, service marketing, and organisation design. Traditional business process modeling [15] most often concentrates on the single-actor case and does not relate business process design decisions to a commercial value viewpoint. Our approach, instead, focusses on processes that are jointly executed by at least two actors in a value network.

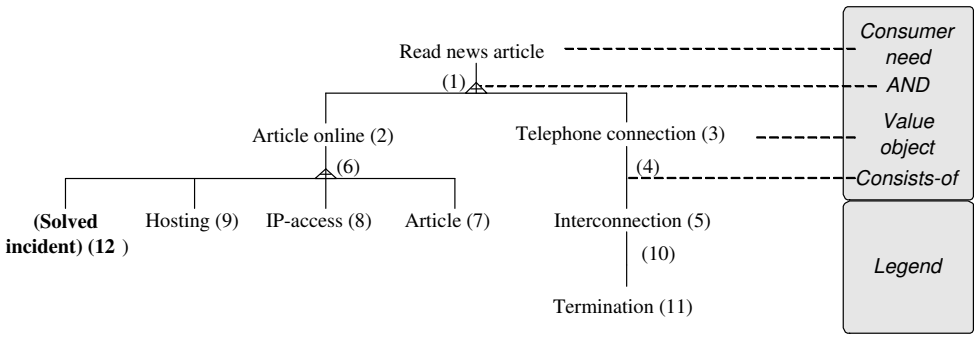


Figure 8. Value object hierarchy extended with the *solved incident* object.

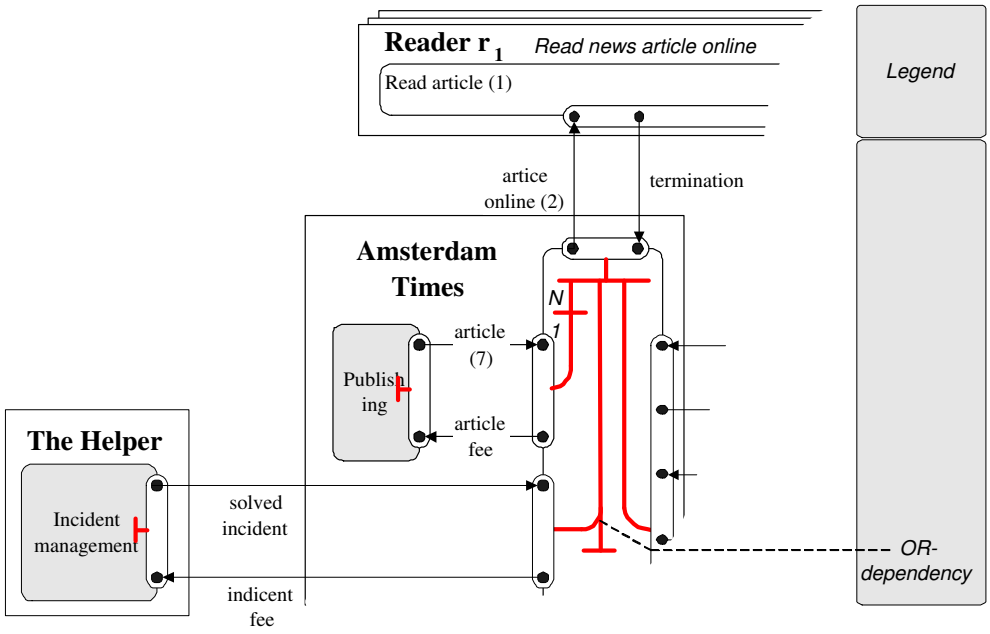


Figure 9. Fraction of a value network showing that *Amsterdam Times* outsources the value activity *indicent management* to an actor called *Helper*.

Key features of the transition from the value viewpoint to the process viewpoint in this paper are

- the interpretation of value transactions in a value network as service delivery and
- the identification of all process steps of both the service provider as well as the consumer to deliver the service.

This technique is comparable with a traditional technique for service design in marketing, called service blueprinting [20]. Service blueprinting is a technique that describes all process steps that are carried out by a service provider and its customer to deliver a service. These process steps are classified in four categories: customer process steps, process steps of provider employees with whom the customer has direct (i.e., face to face or telephone) contact, process steps of provider employees that are not visible to the customer, and support process steps that do not directly related to the customer. Service blueprints resemble activity diagrams, with a swimlane for each of the four categories. For electronic services, a traditional service blueprint is not very useful, as the distinction between the three kinds of provider process steps is not very relevant for an electronic service.

Organisation design theory [5] may provide alternative approaches for the transition from the value viewpoint to the process viewpoint. For instance, processes can be clas-

sified according to Mintzberg's five organisational components [14]: top-level management, middle management, technical support staff, administrative support staff and operational core. Such an approach may provide more insight into the organizational embedding of a new e-commerce idea. However, such an approach pays less attention to the interaction between actors in a value network than the approach presented in this paper.

For the allocation of tasks to information systems (part of the information system viewpoint) there are some practical guidelines [18], some of which date back to Information Engineering [13], but again the integration with the process and commercial viewpoints is weak or non-existing. Our future research will focus on providing guidelines for these transitions, and validating these in additional consultancy projects.

Acknowledgements

We wish to thank the *Amsterdam Times* project consortium for their coorporation and the opportunity for us to learn from the project. This work has been partly sponsored by the Stichting voor Technische Wetenschappen (STW), project VWI.4949, EU-IST project IST-2001-33144 Obelix and EU-EESD project NNE5-2001-00256 BusMod.

References

[1] G. Alonso, F. Casati, H. Kuno, and V. Machiraju. *Web Services: Concepts, Architectures and Applications*. Springer Verlag, 2004.

[2] A. Ankolenkar, M. Burstein, J. R. Hobbs, O. Lassila, D. L. Martin, D. McDermott, S. A. McIlraith, S. Narayanan, M. Paolucci, T. R. Payne, and K. Sycara. DAML-S: Web service description for the semantic web. In *LNCS 2342, Proceedings of The First International Semantic Web Conference (ISWC)*, pages 348–363. Springer Verlag, 2002.

[3] S.-Y. Choi, D. O. Stahl, and A. B. Whinston. *The Economics of Doing Business in the Electronic Marketplace*. MACMillan Technical Publishing, Indianapolis, IN, 1997.

[4] F. Curbera, M. Duftler, R. Khalf, W. Nagy, N. Mukhi, and S. Weerawarana. Unraveling the web services web - an introduction to SOAP, WSDL, and UDDI. *IEEE Internet Computing*, 6(2):86–93, mar/apr 2002.

[5] R. L. Daft. *Organization Theory and Design*. International Thomson Printing, 1998.

[6] ebXML. Business process specification schema v1.01. `http://www.ebxml.org/specs/`.

[7] J. Gordijn and J. M. Akkermans. Designing and evaluating e-Business models. *IEEE Intelligent Systems - Intelligent e-Business*, 16(4):11–17, 2001.

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

[9] J. Gordijn and J. C. van Vliet. On the interaction between business models and software architecture in electronic commerce. In M. Lemoine, editor, *Addendum to the proceedings of the 7th European Software Engineering Conference/Foundations of Software Engineering, Toulouse*, sep 1999. Also available from http://www.cs.vu.nl/~gordijn/.

[10] J. Gordijn and R. Wieringa. A value-oriented approach to e-business process design. In *LNCS 2681, Advanced Information Systems Engineering. Proceedings of the 15th International Conference, CAiSE 2003, Klagenfurt/Velden, Austria*, pages 390–403. Springer Verlag, 2003.

[11] P. Kotler. *Marketing Management: Analysis, Planning, Implementation and Control*. Prentice Hall, 1988.

[12] F. Leymann. *Web Services Flow Language (WSFL 1.0)*. IBM, 2001.

[13] J. Martin. *Strategic Data-Planning Methodologies*. Prentice-Hall, 1982.

[14] H. Mintzberg. *Structure in Fives: Designing Effective Organizations*. Prentice-Hall, 1992.

[15] M. A. Ould. *Business Processes–Modelling and Analysis for Re-engineering and Improvement*. Wiley, 1995.

[16] M. E. Porter. *Competitive Advantage - Creating and Sustaining Superior Performance*. Free Press, 1985.

[17] N. G. Shaw and C. W. Craighead. Technology enablers to recover from failures in e-services. *Communications of the ACM*, 46(6):56–57, 2003.

[18] R. Tagg and C. Freyberg. *Designing Distributed and Cooperative Information Systems*. Thomson, 1997.

[19] D. Tapscott, D. Ticoll, and A. Lowy. *Digital Capital: Harnessing the Power of Business Webs*. N. Brealy Pub., 2000.

[20] V. Zeithaml and M. J. Bitner. *Services Marketing, Third Edition*. McGraw-Hill, 2002.