

Business Modelling is not Process Modelling

Jaap Gordijn¹², Hans Akkermans¹³, and Hans van Vliet¹

¹ Vrije Universiteit Amsterdam

Faculty of Sciences, Vuture.net — Amsterdam Center of e-Business Research

De Boelelaan 1081a, NL-1081 HV Amsterdam, The Netherlands

{gordijn, HansAkkermans, hans}@cs.vu.nl

² Deloitte & Touche Bakkenist, Wisselwerking 46 NL-1112 XR Diemen, The Netherlands

³ AKMC Knowledge Management, Klareweid 19, NL-1831 BV, Koedijk, The Netherlands

Abstract. Innovative e-business projects start with a design of the e-business model. We often encounter the view, in research as well as industry practice, that an e-business model is similar to a business process model, and so can be specified using UML activity diagrams or Petri nets. In this paper, we explain why this is a misunderstanding. The root cause is that a business model is not about process but about value exchanged between actors. Failure to make this separation of concerns leads to poor business decision-making and inadequate business requirements.

1 Introduction

An important part of an e-commerce information system development process is the design of an *e-business model*. Such a model shows the business essentials of the e-commerce business case to be developed. It can be seen as a first step in requirements engineering for e-commerce information systems.

Sometimes, an e-business model is represented using a standard process modelling method such as the UML modelling language (activity diagrams) [11, 3], Petri Nets [13], IDEF₀ [1], STRIM [9] or even (in many practitioner cases) ad-hoc diagrams with some notion of activity. Such models tend to be workflow-oriented: they show the sequence of activities to be performed and sometimes the actors doing so. In addition, it can show branches in a workflow sequence, parallel threads and synchronizations. Thus, a process model shows *how* a particular business case should be carried out.

We do not at all argue that process models are not useful in an e-commerce system development project. On the contrary, a model of the interorganizational business processes is necessary to explain *how* a business model works and results in many requirements for the e-commerce information system to be developed. However, a separation of concerns is needed here. Industry projects (in the telecom, music and energy industries) and case studies we performed [5, 7], show that process models are *not* a good *starting* point for identifying business stakeholder requirements. Most e-business projects start with the *design* of a business model stating *what* is offered *by* whom *to* whom, rather than *how* these offerings are selected, negotiated, contracted and fulfilled *operationally* — as is explained by a process model.

In this paper, we discuss in detail what the differences between business models and process models are. Sec. 2 discusses the various business decisions to be made in

e-business design. Business models and process models refer to different decisions and requirements of different stakeholder groups. For the modelling of business processes well established methods are available, but for the description of an e-business model we find them basically lacking (except for fully informal business-oriented descriptions that leave a big gap with IT system development needs and methods). Therefore, we consider in Sec. 3 what the conceptual structures are that make up an e-business model in general.

This is based on the e^3 -valueTM e-business modelling method we have developed, of which more elaborate explanations and applications are given in [5, 7, 6]. A key point is that an e-business model is built around the notion of *value* networks, a concept absent in process modelling. Thus, business models and process models differ in the decision support they give (i.e., the modelling goals) and in their underlying core concepts. This results in a variety of practical conceptual modelling differences, which are analyzed, illustrated with industry examples, in Sec. 4, and summarized in Sec. 5.

2 Decisions in e-Business Design

The design of e-business applications consists of at least the following processes: (1) the business model design and (2) the business process model design [5]. The design decisions represented by a business model differ from those represented by a business process model. A business model shows the essentials (the strategic intent) of the way of doing business in terms of stakeholders creating and exchanging objects of *value* with each other, while a business process model shows decisions regarding the *operationalization* of a way of doing business.

Business Modelling. Most e-business projects should start with the design of the way of doing business: the business model. Essentially, it provides the design rationale for e-commerce systems from a business point of view.

In our view, the main goal of a business model is to answer the question: “*who* is offering *what* to *whom* and expects *what* in return”. Therefore, the central notion in any business model should be the concept of *value*, in order to explain the creation and addition of value in an multi-party stakeholder network, as well as the exchange of value between stakeholders. The notion of *value* as an important concept in business models is also pointed out in [12] in terms of benefits and revenues.

Consequently, the main design decisions to be represented in a business model are:

1. who are the value adding business actors involved;
2. what are the offerings of which actors to which other actors;
3. what are the elements of offerings.
4. what value-creating or adding activities are producing and consuming these offerings;
5. which value-creating or adding activities are performed by which actors.

Business Process Modelling. A business model does not state *how* value-creating activities are carried out. This is an important goal of business process modelling. Other goals of business process modelling are [9, 13]:

- creation of a common approach for work to be carried out;
- incremental improvement of processes (e.g. efficiency);
- support of processes by workflow management systems;
- analysis of properties of a process (e.g. deadlock free);

To present the *how*, a business process model typically shows the following design decisions:

1. who are the actors involved in the operations;
2. which operational activities can be distinguished;
3. which activities are executed by *which* actors;
4. what are the inputs and outputs of activities;
5. what is the sequence of activities to be carried out for a specific case;
6. which activities can be carried out in parallel for a specific case.

Nature of decision support differs. Accordingly, the nature of design decisions to be represented in an e-business model differs from the decisions to be represented in a process model. An e-business model shows the *what* aspects: what objects of value are created for whom and by whom in multi-party stakeholder network, whereas a business process model shows the associated *how* aspects. An important general goal of conceptual modelling is to provide support for decision-making. Business models and process models thus clearly differ in the types of decisions they are able to support. The importance of separating the *how* from the *what* concerns is anathema already for a long time in conceptual modelling, and it continues to be valid in e-business modelling as ever.

3 Conceptual Structures in e-Business Modelling

A practical business example. First, we will present a practical business example to explain what conceptual structures make up an e-business model. This business illustration is based on a real-life e-business project we carried out, and is about an e-contact service.

The Ad Association is a company that coordinates more than 150 local free ad papers called FAPs. FAPs produce traditional, 'analogue' papers with ads. They are independent, often privately owned organizations, which are located around the world. A FAP serves a geographical region, for instance a large city or a county, because most goods offered in ads only reach a regional market. However, the Ad Association expects that *contact ads* may have a broader scope, even world-wide. Therefore, the Ad Association and the FAPs have decided to exploit their already locally known brand names to set up a contact ad service with a world-wide scope. Moreover, such a service will only be available as an Internet service; contact searchers can submit an ad using their browser, and can search in the ads database via their browser.

In an e-business model we represent decisions regarding stakeholders participating in a business (in this case the e-contact service), and the creation, exchange and consumption of value in such a multi-actor stakeholder network. Figure 1 shows a high-level business model for the contact ad business idea described above. We note that this is only one of the possible business models, and the design of and choice between several

possible strategic alternatives is part of any e-business project. This model shows that contact searchers, a number of FAPs and the Ad Association are involved. More importantly, it represents decisions regarding *who* is exchanging *what* with *whom* and expects *what* in return. For instance, a contact searcher is prepared to submit an ad (and thereby giving up some privacy), and expects a desired contact in return.

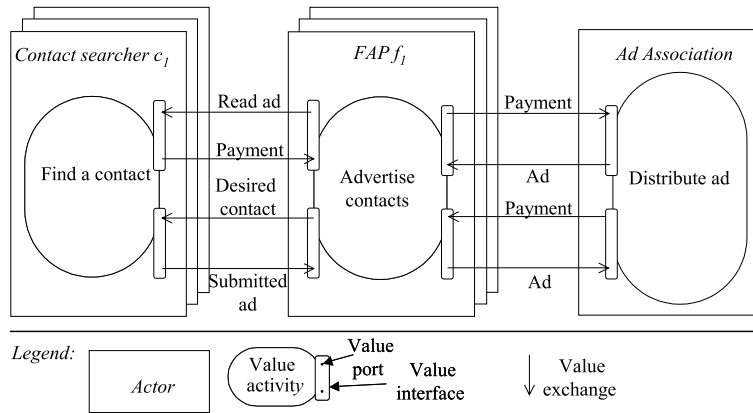


Fig. 1. A top-level business model for the Ad Association.

Generic concepts underlying an e-business model. Analysing business models for various applications such as the above contact ad one, it is possible to define a set of generic concepts and relationships that together make up an e-business model. This is depicted in Fig. 2. It forms part of our *e³-valueTM* method for e-business modelling, which is more extensively described in [5, 7, 6], the latter specifying an ontology for e-business models. Here, we briefly survey the core concepts, only to the extent necessary to analyse the difference between business and process models.

Actor. An actor is perceived by its environment as an *independent economic* (and often also legal) entity. By doing *value activities* (see below) actors add value. In a sound, viable, business model *every* actor is capable of adding value. We distinguish two types of actors:

- business actors;
- end-consumer actors.

These actors differ in the way we calculate their addition of value. Business actors such as shops, resellers and producers buy objects of value (the inputs), perform value-adding activities and produce objects with a higher value (the outputs). The value addition to a business actor is determined by subtracting the price received for the outputs from the price paid for the inputs.

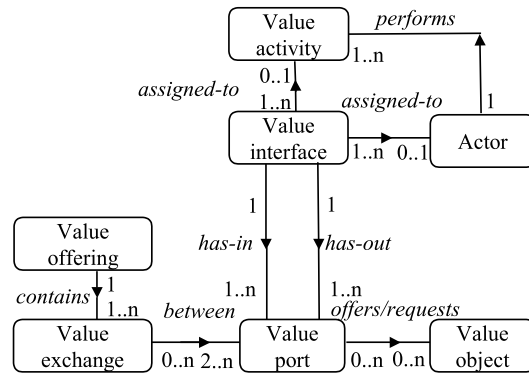


Fig. 2. Generic concepts and relationships underlying an e-business model.

End consumers do not resell the objects of value they buy (otherwise they would be business actors). Rather, they consume the objects of value. An end-consumer actor creates value by consuming an object of value if the *utility* s/he assigns to the consumption experience is higher than the utility associated with the amount of money s/he pays for the object of value, in addition to other resources s/he needs (e.g. time spent) for the consumption experience. The way utility should be measured (e.g. in money) is not trivial and case dependent. As an example, in [4] we present a way to determine the value addition of the consumer experience *listen once to a track of music* to an end consumer. As another example, in [2] we demonstrate how the notion of comfort in the home can be measured in utility terms, as the basis for an e-service.

Value Activity. A value activity is *performed* by an actor to produce objects of value (outputs) by adding value to other objects of value (inputs). Value activities *must* add value. The rationale for this is that during business model design, we want to study various possibilities for the assignment of value activities to actors. Actors, however, are only interested in performing activities if they add value to them.

Value Object. Actors and value activities exchange value objects. A value object is a service, thing, or consumer experience that is of value to one or more actors. A value object is the basic building block for the creation of an offering of an actor to another actor. A value object has one or more valuation properties. Such a property has a name and a unit that indicates the scale in which the valuation is expressed. These properties are used by end-consumers to determine the value they assign to an object.

Value Port. An actor or value activity uses a value port to provide or request value objects to or from its environment. Thus, a value port is used to interconnect actors in a component-based way. Value ports *offer* or *request* value objects.

Value Interface. Actors or value activities have one or more value interfaces modelling the offering of an actor or value activity to its environment. A value interface groups value ports. It shows the value objects an actor is willing to exchange in return for other value objects via its ports. The concept of value interface is based on the principle “one good turn deserves another”.

A value interface has a *valuation function*. It expresses, given valuation properties of objects of all in-ports, the required valuation properties of objects on all out-ports, and vice versa. In other words, a valuation function shows the utility of a value exchange as perceived by an actor: whether an actor is willing to exchange value objects in return for other value objects. The valuation of objects depends on a specific actor evaluating the various dimensions contained in a valuation function [8]. How to do this in quantitative utility terms is investigated in [4].

A value interface *is assigned to* one actor or value activity, and it *has* one or more *in* value ports and it *has* one or more *out* value ports. A port always belongs to exactly one value interface. Multiple value interfaces can be assigned to an actor or value activity. If an actor or value interface has multiple value interfaces, s/he is offering different services to the environment. The value interface says nothing about the time ordering of objects to be exchanged on its ports. It simply states which value objects on which ports are available, in return for some other value objects.

Value Exchange. A value exchange represents the trade of a value object between value ports. It shows which actors are willing to exchange objects of value with other actors. The value ports involved in a value exchange are represented by the *between* relation. At least two value ports participate in a value exchange. A value port can be in multiple value exchanges. A value exchange occurs between ports of opposite direction. A value object flows from an out-port to an in-port. Therefore, at least one in-port and one out-port should be present in a value exchange.

Conceptual constructs differ. The above constructs provide, from a conceptual modelling viewpoint, the major building blocks for designing an e-business model. Clearly, these constructs differ in significant ways from the workflow-oriented ones in process models.

4 Differences between e-Business Modelling and Business Process Modelling

E-business modelling differs in several ways from business process modelling:

1. The goal of e-business modelling is to come to a common understanding between stakeholders regarding *who* is offering and exchanging *what* with *whom* and expects *what* in return. The goal of a process model is to clarify *how* processes should be carried out, and by *whom*.
2. The concepts in e-business modelling are centred around the notion of *value*, while in process modelling concepts focus on *how* a process should be carried out.

3. The statements about the Universe of Discourse differ in e-business models and process models. An e-business model says to which extent actors add value and whether actors are willing to exchange objects of value with each other. A process models states which activities should performed, in which order, and which objects (in which order) should be exchanged.
4. Different model decomposition rules apply. In e-business modelling we use decomposition of value-adding activities as a way to discover new value-adding activities, for instance to discuss new alternative assignments of such activities to actors. Decomposition of activities in process modelling serves the goal of clarity, or studying various resource allocations (e.g. operational actors) to activities.

The first two differences have been discussed in the previous sections. The latter two will be elaborated in detail in this section. Here, we will take many practical examples from the contact ad case discussed in Sec. 3. For this business case example, Fig. 1 shows an e-business model, and Fig. 3 introduces an activity model. Detailed views are given in Fig. 4 and Fig. 5.

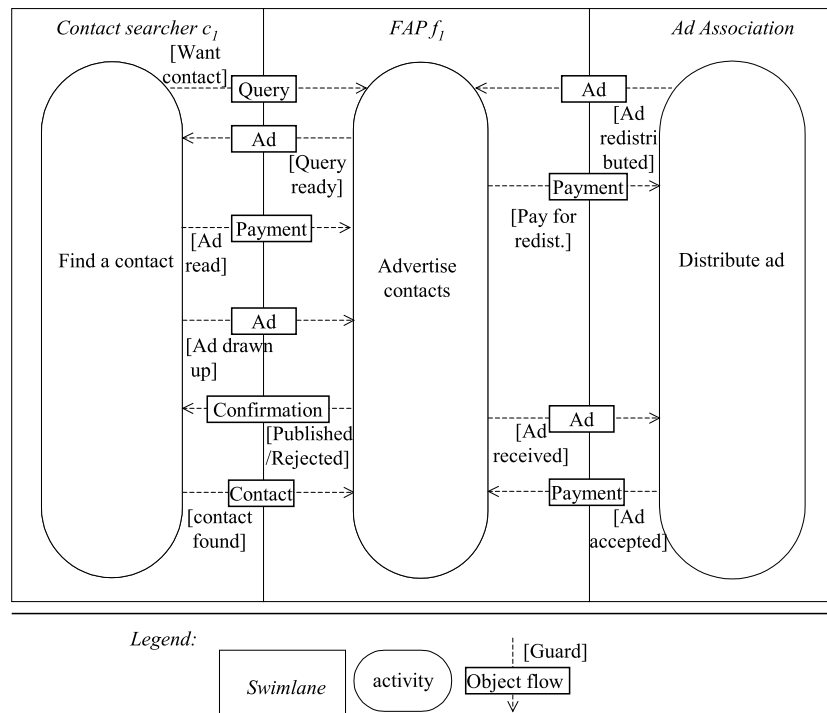


Fig. 3. A top level activity model for the Ad Association.

4.1 Value Object and Object

The notion of value. In a business model, objects are only shown if they are of *value* to stakeholders. In a process model, objects are shown if they serve as required inputs of activities or are produced as outputs. As a consequence, not all objects that are part of a process model need appear in a business model, because some objects may not be of direct value to someone. This is for instance the case for objects that represent control information, necessary to steer the process. Also, a business model may identify objects that are not present in a process model. This is for instance the case if an object indicates a valuable consumer experience, while no direct transfer of goods, services or information is involved.

Example: An object present in a process model, not present in a business model. Fig. 3 shows an object *confirmation*, which models that a contact searcher receives a (positive or negative) confirmation after a submission of an ad. This *confirmation* object is not modelled in the business model (Fig. 1), because it is not of direct value to the contact searcher. It is only needed as control information, for instance to trigger the contact searcher to re-submit his/her ad after rejection.

Example: An object in a business model, not present in a process model. In Fig. 3 a *desired contact* is not present, because there is no corresponding physical or information object flowing from the FAP to the contact searcher. A desired contact states a consumer experience of the contact searcher; namely that s/he found a contact s/he likes. As a valuable consumer experience, it is present in the business model (Fig. 1).

Object properties. Different subsets of object properties are identified for business models and process models. A business model identifies those object properties, which can be used by a stakeholder to determine the *value* of the object, whereas object properties in process models can be used by an activity to determine a *state transition*.

Example: state transition property and value property. The *publishing date* of an ad on a website is a property useful in a process model, because it can be used to determine a state transition; from an invisible ad to a visible ad on sites of FAPs. This property, however, is not very useful in determining the *value* of an ad; neither to the FAPs nor to the contact searcher. Because the business model in Fig. 1 states that a reader has to pay for reading an ad, an interesting value property is the *likelihood* an ad contains a contact the reader is interested in (e.g. based on the reader's personal profile). Such a property partly determines the value a reader assigns to an ad read.

Objects and attributes differ. In sum, objects themselves as well as the kind of object properties differ between a business model and a process model. In a business model objects need to yield value to someone, while in a process model objects serve as inputs and outputs for activities. In a business model, properties of an object should be usable for valuing the object by an actor, whereas in an activity model properties can be used to determine a state transition.

4.2 Value Exchange and Flow

Transfer of ownership. As explained in Sec. 3, objects of value are exchanged between actors/value activities through *value exchanges*. The goal of such a construct is to model a *legal transfer* of value objects, for instance a transfer of ownership (e.g. goods, money), an obligation to realize a consumer experience, or a grant to exercise a right or license (e.g. for music or software). In a process model data flows and control flows are used to model a transition from one activity to another. It is used to express *how* activities should be carried out in terms of sequences or parallelisations of activities.

Example: Flows and rights. The process model in Fig. 3 contains a data flow from the contact searcher to the FAP called *contact* that states that the contact searcher reports the experience of a desired contact to the FAP. This flow relates to the *desired contact* value exchange (Fig. 1) from the FAP to the contact searcher, but it is not the same. The *contact* flow is necessary as *control* information, for instance as a trigger to remove a published ad as soon as a desired contact occurs, while the *desired contact* value exchange models the *valuable experience* itself. Note that in the process model, the confirmation flows from contact searcher to FAP, while the desired contact flows from the FAP to the contact searcher.

No direct physical or information flow. A value exchange may coincide with a flow of a physical product or information if these are of value to a stakeholder. However, sometimes a value exchange states a consumer experience, which has no underlying direct physical or information flow.

Example: No physical flow. The previous example illustrates an exchange of value, a desired contact, which represents a consumer experience, with no associated direct physical or information flow.

Model state changes differ. In conclusion, a value exchange expresses a change of ownership (as an economic result, not as a process outcome), which is normally not expressed in process models. Moreover, some value exchanges do not imply a physical or information flow directly, but instead express an actors' consumer experience.

4.3 Value Interface

In a business model, we have the notion of *value interface* expressing the principle "one good turn deserves another" (a rule or law of value exchange). This allows stakeholders to clarify to each other *what* objects of value they are prepared to exchange in return for other objects; a key decision during business modelling. Such a principle is not present in process models.

Example: One good turn deserves another. From Fig. 3 it cannot easily be concluded that a reader has to pay for reading an ad, while Fig. 1 clearly shows that a *read ad* is offered in return for a *payment*.

4.4 Activity

Value adding. In process modelling, an activity denotes something to be done, in order to produce outputs as a result of inputs and resources. In a business model, we distinguish activities only if they *add value* for the performing stakeholder. The rationale for introducing value-adding activities is that, for a particular business case, we want to determine the amount of value addition for each actor, and we want to address and discuss various assignments of activities to stakeholders. Stakeholders are only interested in performing activities if these add value.

Decomposition. The different interpretations of the *activity* concept in business models and process models leads to different decompositions.

In the literature on process modelling, a number of motivations are given for the decomposition of activities into sub-activities. IDEF₀ [1] indicates that an activity should be recursively decomposed in 5 to 7 sub-activities, until a common understanding about the activity is reached by stakeholders. In this case, decomposition serves the goal of *clarity*. In STRIM [9], activities are decomposed until they can be regrouped and assigned to a particular role (i.e., operational actor). Decomposition then serves the goal of clarifying resources needed in carrying out tasks. In a business model, however, we only decompose a value activity if *all* resulting sub-activities themselves add value. In [10] and [12], this is referred to as value chain deconstruction, as a way to discover new activities which can be successfully assigned to alternative commercial actors.

Example: Different decompositions. Fig. 5 decomposes the activities introduced in Fig. 3. For brevity, we do not show the complete decomposition but focus on the submission of an ad. The main goal of the decomposition is to illustrate *how* a submission process should be carried out. After an ad is submitted by a contact searcher, it is checked (e.g. for absence of dirty language). If the ad passes this check, it is added to the website of a FAP and the contact searcher receives a confirmation. Also, the ad is offered to the Ad Association, which pays for it afterwards. The Ad Association supplies the ad to other FAPs. In sum, this detailed process model shows the activities necessary for a submission, as well as their execution sequence and parallel threads for an ad submission.

Figure 4 shows a decomposition of the value activities in Fig. 1 into *profitable* sub-activities. The decomposition operation is defined as follows:

1. a value activity can be decomposed in other (sub) value activities if each sub-value activity adds value;
2. consider for each pair of sub-value activities new value interfaces and value exchanges if required.

The *find contact* value activity is decomposed into two sub-value activities: (1) *read an ad*, and (2) *submit an ad*. Both activities are likely to add value for contact searchers; they both enhance the chance to find a desired contact. The *advertise contacts* value activity is decomposed into three sub-value activities. Note that between these sub-value activities new value exchanges have been introduced. Also note that the *check an ad* activity is considered to add value; this is especially the case when a FAP requests another FAP to check an ad, for instance if a FAP does not speak the language in which an ad was written.

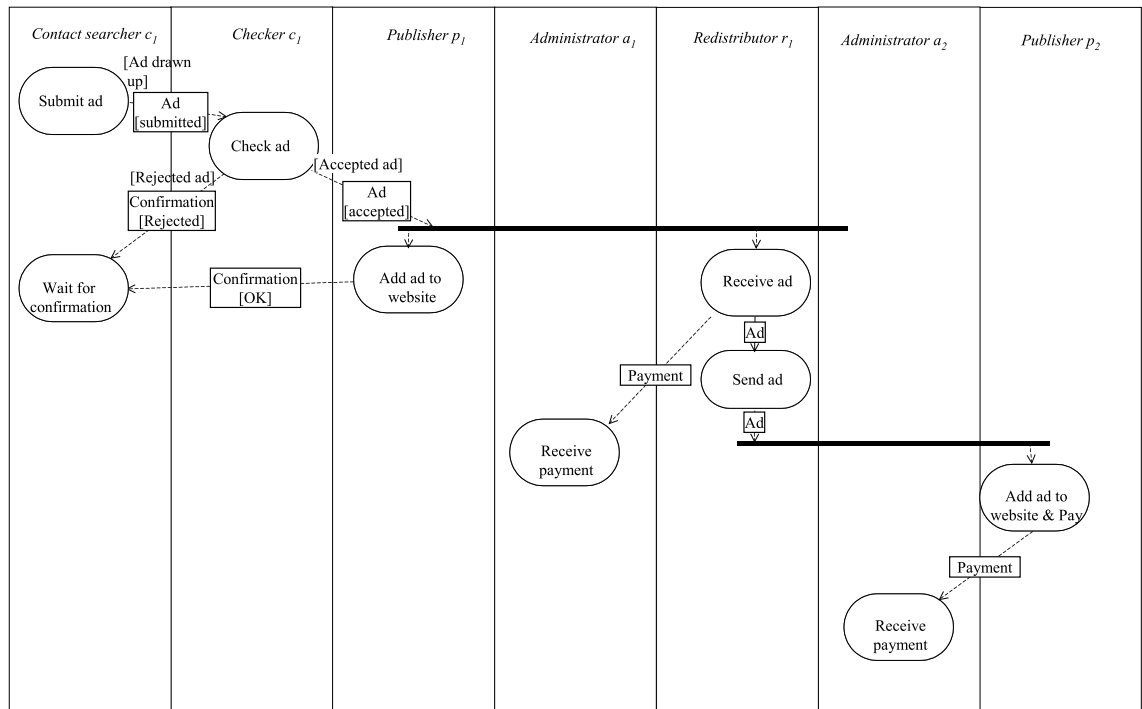


Fig. 5. A decomposed process model for the Ad Association.

Model decomposition rules differ. In sum, in a process model, decomposition is often led by the motivation to show a process flow in detail, while in a business model it is led by a search for commercially viable sub-activities.

4.5 Actors

Individual actors. In a process model, the actor itself is usually not shown at the instance level. At most it is indicated that a number of actors capable of performing a particular activity, should be present, for instance to model resource management. When designing business models, it should be possible to identify the profitability of a business model to a particular actor. During business modelling, these individual actors are important stakeholders. Therefore, in a business model, actors sometimes are mentioned on an individual basis.

Value Adding. Actors in an process model are indicated for purposes such as resource allocation and scheduling. However, in a business model we distinguish actors to facilitate reasoning about *value addition*. Therefore, actors are not individual agents performing activities, but economic and legal entities that engage in business transactions.

Example: Operational actors and commercial actors. In Figure 5, actors performing activities are represented by swimlanes. The actual actor instances are not mentioned, while the business model (Fig. 1) indicates the existence of a number of FAPs which can be addressed on an individual basis. Moreover, in the business model we distinguish FAPs, being legal entities that engage in business transactions, whereas in the process model we identify resources carrying out work for such an entity, such as a checker, a publisher, a redistributor, and an administration officer.

5 Conclusions

e-Business modelling and process modelling are both forms of conceptual modelling, both are necessary for good e-business design, but they differ in several significant ways. First of all, the main goal of e-business modelling is to reach agreement amongst stakeholders regarding the question “*who* is offering *what of value* to *whom* and expects *what of value* in return”. In contrast, an important goal of process modelling is to reach a common understanding about *how* activities should be carried out (e.g. in which order). These are two different modelling goals, asking for different modelling methods with different constructs. Modelling strategic intent of e-business differs from modelling operational fulfilment.

As a result, the contents of an e-business model and a process model also differ in a number of ways:

1. The concepts in e-business modelling are centred around the notion of *value*, while in process modelling concepts focus on *how* a process should be carried out in operational terms.

2. In an e-business model, an actor adds value, while in a process model an actor performs an operational process.
3. In an e-business model, objects represent something of value to a stakeholder, while in a process model objects serve as inputs and outputs for activities and may be used to steer the process flow.
4. In an e-business model, object properties can be used by a stakeholder to determine the value of an object. In a process model, object properties are used to determine state transitions.
5. In an e-business model, value exchanges represent a transfer of ownership, while in a process model a flow of information or goods implies a change of state;
6. In an e-business model, we have the notion of “One good turn deserves another”, which is conceptualised by the value interface. Such a notion is absent in process modelling.
7. In an e-business model, we are only interested in activities which are capable of adding value. Decomposition of such activities is done to discover smaller chunks of activities that still add value. Discovering these activities often leads to re-assignment of activities to actors. In a process model, decomposition serves the goal of clarification of the workflow or to show the assignment of activities to working actors. Hence, the model decomposition rules are different.

We have used our e^3 -valueTM method in a number of industrial e-business development projects. Our experience is that a focus on the creation and distribution of value in a stakeholder network is a convenient way to express, negotiate and clarify business models to stakeholders. Distinguishing the distribution of value (e-business modelling) from the way processes are actually performed (process modelling) leads to a separation of concerns of stakeholders and clarifies the discussions.

In our ongoing research, we are developing tools to implement our e-business model concepts, to formally represent business models and to check whether they are well-formed. Secondly, we are enhancing our method with (quantitative) ways to determine the value of objects. Especially for end consumers this is a multi-faceted and open-ended issue [8, 4]. Thirdly, we are specializing our concepts for use in the digital content industry.

Acknowledgement. This work has been partly sponsored by the Stichting voor de Technische Wetenschappen (STW), project nr VWI.4949.

References

1. *IDEF₀* method report. http://www.idef.com/Complete_Reports/idef0, 1981.
2. E. Boertjes, J.M. Akkermans, R. Kamphuis, and R. Gustavsson. Agents achieving customer satisfaction — the COMFY comfort management system. In *Proceedings 5th Int. Conf. on the Practical Application of Intelligent Agents and Multi-Agent Technology PAAM-2000 (Manchester, UK, 10-12 April 2000)*, pages 75–94, Blackpool, UK, 2000. The Practical Application Company Ltd.
3. Martin Fowler and Kendall Scott. *UML Distilled - Applying the standard object modelling language*. Addison Wesley Longmann, Inc., Reading, Massachusetts, 1997.

4. J. Gordijn, J.M. Akkermans, and J.C. van Vliet. Selling bits: A matter of creating consumer value. In *First International Conference on Electronic Commerce and Web Technologies ECWEB-2000 (Greenwich, UK, 4-6 September 2000)*, 2000. Submitted. Available from <http://www.cs.vu.nl/~gordijn>.
5. J. Gordijn, J.M. Akkermans, and J.C. van Vliet. Value based requirements creation for electronic commerce applications. In *Proceedings of the 33rd Hawaii International Conference On System Sciences (HICSS-33)*, pages CD-ROM. IEEE, January 4-7 2000. Also available from <http://www.cs.vu.nl/~gordijn>.
6. J. Gordijn, J.M. Akkermans, and J.C. van Vliet. What's in an electronic business model. In *12th International Conference on Knowledge Engineering and Knowledge Management EKAW-2000 (Juan-les-Prins, 2-6 October 2000)*, 2000. Submitted. Available from <http://www.cs.vu.nl/~gordijn>.
7. J. Gordijn, H. de Bruin, and J.M. Akkermans. Integral design of E-Commerce systems: Aligning the business with software architecture through scenarios. In H. de Bruin, editor, *ICT-Architecture in the BeNeLux*, 1999. Also available from <http://www.cs.vu.nl/~gordijn>.
8. Morris B. Holbrook. *Consumer value: a framework for analysis and research*. Routledge, New York, 1999.
9. Martyn A. Ould. *Business Processes - Modelling and Analysis for Re-engineering and Improvement*. John Wiley & Sons, Chichester, England, 1995.
10. M.E. Porter and V.E. Millar. How information gives you competitive advantage. *Harvard Business Review*, pages 149–160, 1985.
11. James Rumbaugh, Ivar Jacobson, and Grady Booch. *The Unified Modelling Language Reference Manual*. Addison Wesley Longmann, Inc., Reading, Massachusetts, 1999.
12. Paul Timmers. *Electronic Commerce: Strategies and Models for Business-to-Business Trading*. John Wiley & Sons Ltd., Chichester, England, 1999.
13. K.M. van Hee. *Information Systems Engineering - A formal approach*. Cambridge University Press, Cambridge, 1994.