

## From Business to Process Models – a Chaining Methodology

Birger Andersson<sup>1</sup>, Maria Bergholtz<sup>1</sup>, Bertrand Grégoire<sup>2</sup>, Paul Johannesson<sup>1</sup>,  
Michael Schmitt<sup>2</sup>, Jelena Zdravkovic<sup>1</sup>

<sup>1</sup>Department of Computer and Systems Sciences  
Stockholm University and Royal Institute of Technology  
Forum 100, SE-164 40 Kista, Sweden  
{ba, maria, pajo, jzc}@dsv.su.se

<sup>2</sup>Public Research Centre Henri Tudor  
29, Avenue John F.Kennedy  
L-1855 Luxembourg-Kirchberg, Luxembourg  
{Bertrand.Grégore, michael.schmitt}@tudor.lu

**Abstract.** In this paper we discuss the problem of how to go from a business model to a process model in a systematic way. Business models are economic models used for business analysis, while process models capture low-level business activities and their coordination. We propose a method that starts with a business model where the main actors and their relationships are identified. This forms a basis for design of a final process model. Processes are described in terms of patterns stored in a pattern library.

### 1 Introduction

In e-commerce it is possible to identify two basic types of models: business models and process models. A business model focuses on the *what* in an e-commerce system, identifying agents, goals, resources, and exchanges of resources between agents. Thus, a business model provides a high-level view of the activities taking place in e-commerce. A process model, on the other hand, focuses on the *how* in an e-commerce system, specifying operational and procedural aspects of business communication. Although different, business and process models are interrelated as the correctness and quality of a process model depends on its ability to support a business model. Two questions arise in the context. First, how can we be sure that an operational process does support the goals and exchanges specified in a business model? This question has been discussed in [1]. The authors of that work do not specify how to derive a process model from a value model, but provides a correctness criterion for a given process model with respect to a value model. A second question is: how can we systematically construct a process model starting from a given business model?

We propose a method for addressing the second question by starting from a business model and arriving at a process model by using patterns from a process library.

As suggested in [2], the method takes into account communicative, logistical, and trust aspects, which to a large extent determine the design of a process model. The method is based on two recent contributions in the areas of business modelling and requirements engineering - an analysis of the notion of value objects [3] and the use of process patterns [4]. In [4] a structured query session is proposed as a method. In [5] a model of different types of dependencies is introduced as an intermediate step between the value and the process model. In [6] additional aspects, such as risk mitigation, are considered and a pattern based approach is chosen when the final process model is designed.

The business model formalism that we use in this paper is that of e3-value as proposed in [7]. The formalism is based on identifying actors involved in a business as well as value objects that those actors exchange. The method is based on a number of phases of business transactions. Open-EDI [8] introduces five phases of business transactions: the *planning* and *identification* phases where values and agents are identified respectively, the *negotiation* phase where commitments to particular value exchanges are established, the *actualization* phase where the agreed value exchanges are carried out and the *post-actualization* phase where possible complaints are performed.

In the next section we describe our method which is followed by a deep analysis of the use of process patterns in Section 3. Section 4 contains some concluding remarks and future research.

## 2 Chaining Method

In order to design a process model supporting a business model, we need to analyze in more detail the structure of value exchanges and value objects [3]. A first distinction can be made between resources and rights on resources. A *Resource* is an object that is regarded as valuable by some actors. A *Right* on a resource means that an actor is entitled to use that resource in some way. An example is the ownership of a book, which means that an actor is entitled to read the book, give it to someone else, or even destroy it. Another example of a right is borrowing a book, which gives the actor the right to read it, but not to give it away or destroy it. For a value exchange, both the resource being transferred and the right on the resource must be specified. For example, the value exchanges in which a car is sold and borrowed respectively, concern the same resource but transfer different rights on that resource.

Another component of a value exchange is transferring the custody of the resource being exchanged from one actor to another. An actor has the *Custody* of a resource if he has immediate charge and control of the resource, typically physical access of the resource. If an actor has the custody of a resource, this does not mean that she has any rights on the resource. For example, a distributor may have the custody of some goods, but he is not allowed to use the goods in any way. Providing custody of a resource is essential in a value exchange, as the buyer is typically unable to exercise the rights she gets unless she has custody of the resource.

A value exchange may also include the transfer of some evidence document that certifies that the buyer has certain rights on a resource. A typical example of an evi-

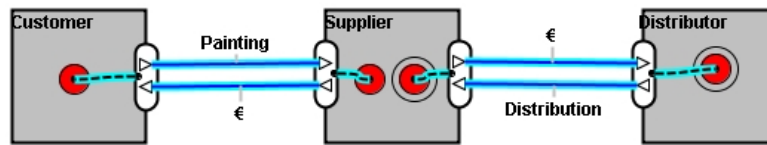
dence document is a movie ticket that certifies that its owner has the right to watch a movie. Summarising, a value exchange can be seen as combining four components:

- The resource being exchanged from one actor to another, e.g., a book
- The right that the buyer obtains on the resource, e.g., the ownership of a book
- The custody of the resource, e.g., the delivery of a book to the buyer
- The evidence document, e.g., a ticket

While the first two components, the resource and right, always are included in a value exchange, the last two components are optional. For example, when buying a piece of land, the buyer is typically not given the custody of that resource. Clearly, evidence documents are not always provided in a value exchange. Furthermore, the provision of custody and evidence documents may be so trivial that it is not of interest to make them explicit.

The analysis above will give a basis for the initial steps of the proposed methodology. The starting point is an e3-value model, which we extend by introducing elements corresponding to the components of value exchanges. In Fig. 1 rectangles represent actors and arrows represent value exchanges.

**Step 1:** Consider an e3-value model (see Fig. 1).



**Fig. 1.** The e3-value model.

**Step 2:** For each value exchange, determine whether the custody component of the value exchange exists and shall be modelled explicitly. If so, add one or more arrows to the model that represent transfers of custody from one actor to another.

This step can be viewed as “factoring out” the custody component of a value exchange and modelling it explicitly by additional flows in the model. It should be noted that several actors may be involved in transferring the custody from one actor to another. This is illustrated in the extended model in Fig. 2a, where the custody of a painting is first transferred from the producer to the distributor and then from the distributor to the customer (flows for custody are dashed).

**Step 3:** For each value exchange, determine whether the evidence document component of the value exchange exists and shall be modelled explicitly. If so, add one or more arrows to the model that represent transfers of evidence documents from one actor to another.

Analogously to the step for custody, this step can be viewed as factoring out the evidence document component of a value exchange. Also in this case, several actors may be involved, e.g., when a ticket office supplies tickets on behalf of other service providers. In Fig. 2b, the transfer of the owner certificate of a painting is made explicit (flows for evidence documents are dotted).

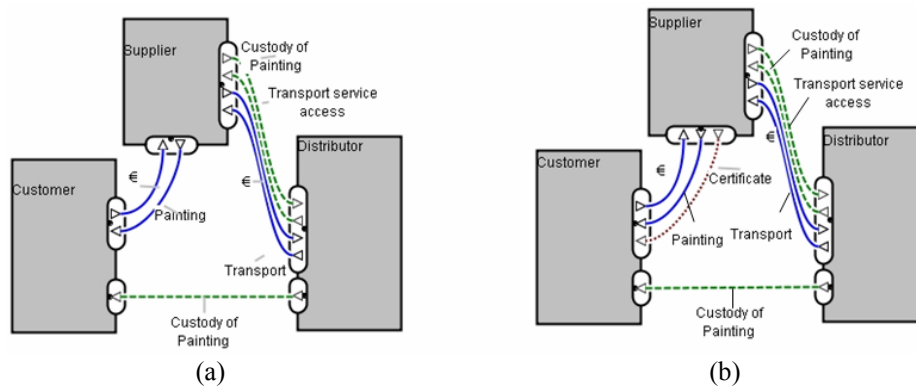


Fig. 2. e3-value model extended with (a) Custody flow and (b) Evidence flow.

**Step 4:** Identify a set of processes based on the extended e3-value model from Step 3 and the Open-EDI transaction phases described in Section 1.

- For each value transaction, one negotiation process is added
- For each arrow in the extended model, one actualization process is added
- For each arrow in the extended model, optionally one post-actualization process is added

The following processes will be identified based on the extended e3-value model in Fig. 2 (The nomenclature is Custody when discussing physical goods and Access when discussing services): Negotiation process for Painting, Negotiation process for Transport, PaintingCustody2Distributor, PaintingCustody2Customer, TransportAccess2Supplier, Money2Supplier, Money2Distributor, Painting2Customer, Transport2Supplier and EvidenceDoc2Customer.

The above steps only concern the identification of flows relating the actors, which gives a set of processes. The following step concerns the design of the resulting processes. For this end we make use of a library where processes are described in terms of patterns. A pattern contains information about properties that characterize sets of processes. For example, a pattern may characterize a process as involving three actors and to be secure.

**Step 5:** For each process, select a pattern based on the resource managed by the process and the goals of the actors (for details on this selection procedure, see the next section). Apply the selected pattern to the set of identified processes.

Applying a pattern to the set of identified processes will result in specifying the internal structure of a process or relating processes to each other. In some cases, a process is not retained but replaced by two or more new processes related to one or more added actors. For example, a payment process between a buyer and a seller can be restructured to include an intermediate additional actor (a bank) that participates in new processes with both the buyer and seller. Application of patterns with this effect means that the structure of the business model is also changed. In the following section we elaborate on the concept of pattern and illustrate the use of a pattern library.

### 3. Patterns

Patterns can be described from different points of view. From a model designer's perspective, they are off-the-shelf business processes that can be linked and chained up to build more complex, end-to-end transactions. The UN/CEFACT Unified Modelling Methodology [9] states that the use of patterns optimizes business process and information model reusability, a precondition for this being that there is a pattern corresponding to the requirements of the business collaboration. Let us consider a sample pattern for a situation of goods' delivery between a supplier and a distributor.

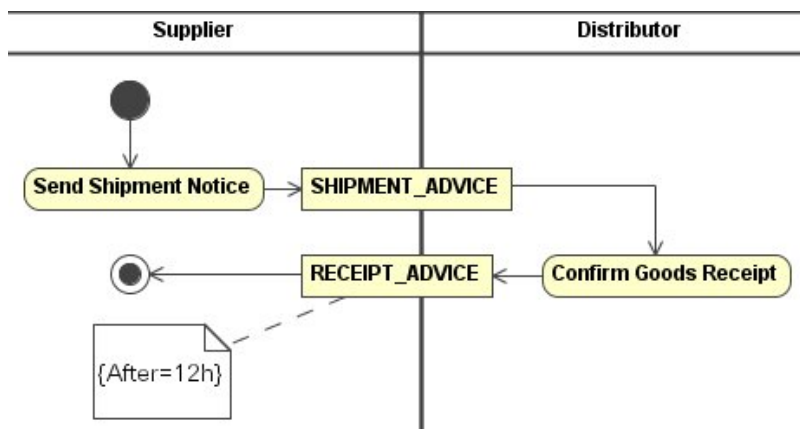


Fig. 3: A pattern component for an activity of goods' delivery

The UML activity diagram in Fig. 3 defines the dynamic aspects of business collaboration. It describes the course of action and fixes a time window within which the distributor must confirm the goods' receipt. (More information on Timed Activity Diagrams can be found in [10]).

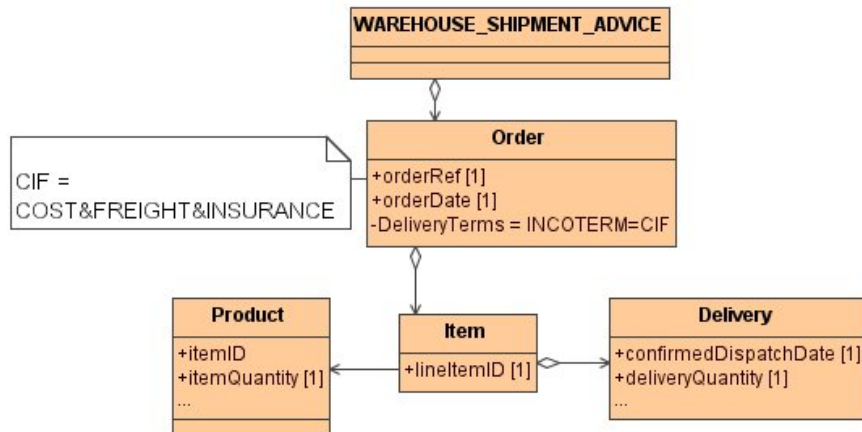
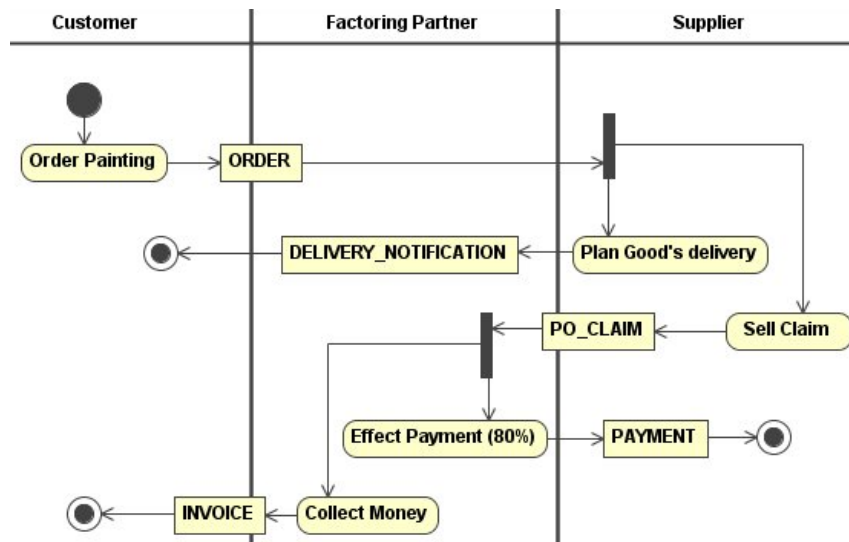


Fig. 4: A pattern component for a SHIPMENT ADVICE document.

The static aspects of the business collaboration, or rather the information content of the business documents, are illustrated in the UML class diagram in Fig. 4. The *SHIPMENT\_ADVICE* document refers to a previous customer order and specifies the exact delivery schedules and quantities for each line item of that order. The order document is used in this context as a point of reference and links the goods' delivery activity to the larger business context of the overall transaction it is embedded in. The chaining algorithm must ensure that relevant business information of the previous collaboration phases are passed on to the delivery phase. Note that the same pattern may be employed for a business case where the distributor would take ownership of the goods, and thus all costs and risks associated with the transport. In this case, it would be sufficient to modify the terms and conditions for the goods delivery in the preceding order document. Patterns are always described by both its dynamic and static properties.

In business terms, patterns can be considered as instruments to achieve business goals. Let us consider the example in Fig. 5:



**Fig. 5:** Flow of information in a factoring business case

A Supplier receives a purchase order from a Customer. Instead of invoicing the Customer after the goods' delivery, he sells his claim on the order value to a Factoring Partner who pays him 80% of the value of the claim's value upfront before he collects the money from the Customer. This pattern solves several business purposes:

- For the Supplier, it fulfils a financing function as he receives his money upfront. It also relieves him from the burden to collect his money from the Customer. Whether there's a delay in payment or the Customer goes bankrupt, he can be sure to have 80% of the total order value.
- For the Factoring partner, he can either make a profit of 20% of the order value or loose 80%. In order to reduce his risk exposure he would want to engage only in the business with the Supplier, if the latter contracts him on a regular basis.

Given such a business case, we can derive the commercial conditions for the use of our Factoring pattern: there need to be a Supplier who has regular trade with a Customer and wants or needs – due to either a requirement for financing or because he perceives the risk of non-payment with regards to the Customer in question as rather high - to mitigate all payment-related risks by involving a Factoring partner. Furthermore, the profit margin must be higher than 20 % in order to obtain an economically sound Factoring solution.

Generally speaking, we can associate all patterns with their business characteristics so as to find the best pattern for any given business situation, and to evaluate the pattern's feasibility with regards to the business goals of the firms involved. There are four categories of pattern characteristics:

- The *collaboration phase* it refers to. Some patterns (e.g. business processes for price negotiations) apply to one collaboration phase of the Open-EDI model, while others are more generic.
- The *type of resource* used in a business exchange. Patterns that can be used to mitigate payment risks are in most cases not suited to deal with information exchange or the transport of physical goods or services.
- *Contractual characteristics*:
  - Patterns have specific characteristics with regards to their *ability to manage risks*. While some patterns put all risks on a single business partner, others do balance risks among the firms involved. An example of the latter is a letter of credit. A specific aspect is the moment (and hence the risks associated with) when the *transfer of title* on a resource passes from the seller to the buyer.
  - Patterns can be described in terms of their *cost allocation* behaviour. Many patterns, among which we find the Incoterms, define which party needs to pay which part of the value creation activity.
- *Process characteristics*. This category subsumes the practical concerns related to the execution of the business process the pattern describes: process efficiency, duration as well as administrative and technical complexity of execution.

For each of these categories, there is a range of possible values, and each vector of values (instances) maps onto a set of patterns that would fit the requirements.

#### 4. Conclusion and Future Work

In this paper we have discussed the problem of how to go from a business model to a process model in a systematic way. A method is proposed that starts with a business model where the main actors and their relationships, in the form of value exchanges, are identified. In a number of steps, each value exchange is analysed and identified as a sub-process that is to constitute the final process model. When analysing value exchanges between business actors, we consider them as consisting of components, which we call resource, right, custody, and document evidence. Each separate component motivates a sub-process in the process model, and consequently a modeller should decide whether to treat a value exchange as a unit or factor out one or several of its components. Using a set of criteria, identified sub-processes are matched with

the complete process descriptions described in form of patterns stored in a library. Use of the patterns enables a fast, module-based process design and stipulates the alignment between the starting business model and the process model.

Future research also involves investigations on how to properly use a pattern library in this context. For instance, should the current content in the library decide the design of the process model, or should the pattern library be used for validation purposes when the modeller is finished, or both. Another topic for future research includes further identifying and describing the dependencies that exist between processes. These dependencies are expected to be useable means of deriving and facilitating the selection of appropriate candidate patterns for implementing a particular relationship, and consequently increase the quality of the resulting process model.

## References

1. Wieringa R.J. and Gordijn J., "Value-Oriented Design of Service Coordination Processes: Correctness and Trust", *ACM Symposium on Applied Computing, Organizational Engineering Track*, 2005.
2. Weigand H., Johannesson P., Andersson B., Bergholtz M., Edirisuriya A. and Ilayperuma T. "Value Object Analysis and the Transformation from Value Model to Process Model". Accepted for publication at E-ISA'06 proceedings. Bordeaux, France. 2006.
3. Weigand H., Johannesson P., Andersson B., Bergholtz M., Edirisuriya A. and Ilayperuma T. "On the Notion of Value Object". Accepted for publication in CAiSE'06 proceedings. Luxembourg, 2006.
4. Jayaweera P. "A Unified Framework for e-Commerce Systems Development: Business Process Pattern Perspective (BP3)", PhD Thesis (ISBN 91-7265-938-6) Sept, 2004.
5. Andersson, B., Bergholtz M., Edirisuriya A., Ilayperuma T., Johannesson P. "A Declarative Foundation of Process Models", *Proc. CAISE'05*, Springer-Verlag, LNCS, 2005
6. Bergholtz M., Grégoire B., Johannesson P., Schmitt M., Wohed P. and Zdravkovic J. "Integrated Methodology for linking business and process models with risk mitigation", *REBNITA 05*.
7. Gordijn J., Akkermans J.M. and Vliet J.C. van, "Business Modeling is not Process Modeling", *Conceptual Modeling for E-Business and the Web*, LNCS 1921, Springer-Verlag pp. 40-51, 2000.
8. UN-Centre for Trade Facilitation and Electronic Business, *Open-EDI phases with REA*, [http://www.unece.org/cefact/docum/download/02bp\\_rea.doc](http://www.unece.org/cefact/docum/download/02bp_rea.doc)
9. UN-Centre for Trade Facilitation and Electronic Business, *Unified Modelling Methodology*, <http://www.untmg.org/>
10. Nicolas Guelfi, Amel Mammari. *A Formal Semantics of Timed Activity Diagrams and its PROMELA Translation*, Asia Pacific Software Engineering Conference (APSEC'05), Taipei, Taiwan, IEEE Computer Society Press, December 2005.