

A Short Comparison of Business Process Modelling Methods under the Perspective of Structure and Behaviour

Elmar J. Sinz^{*,a}

^a Faculty of Information Systems and Applied Computer Sciences, University of Bamberg, Germany

Abstract. EPC, BPMN, SOM and petri nets are methods to business process modelling which look quite different at the first glance. Considering the two main characteristics of a system, structure and behaviour, this short article shows two things: (1) in all methods the behaviour model can be regarded as a petri net enriched with certain semantics, (2) the structure model is missing in all methods besides SOM, thus wasting a lot of semantics.

Keywords. Business Process Modelling • EPC • BPMN • SOM • Petri Net

1 A system view on business processes

A business process can be understood as (1) collection of activities, separated by means of common attributes, (2) event-driven flow of these activities, (3) adoption of inputs and generation of outputs having a value for the consumers, and (4) assignment and utilization of some resources (Ferstl and Sinz 1993; Vossen and Becker 1996). From this follows that a business process can be understood as a system, consisting of components and relationships.

What is a business process under the perspective of structure and behaviour? Structure and behaviour are the main characteristics of a system. Compared with a transportation system, structure is the network of roads connecting the components; behaviour is the traffic on it. The structure of a system determines the scope of its behaviour; a certain behaviour of a system is only possible, if the structure supports it.

As an example, a business process involving a company and a customer is used. The business process starts with product information sent from the company to the customer. In case it shows what the customer wants, the customer places an order

which is returned to the company. Processing the order, the company submits a shipping order to its store, which releases the shipping to the customer. An internal shipping report finishes the business process.

In the following, it is shown that EPC, BPMN and SOM model the behaviour of a business process based on petri nets. Despite looking quite different, the methods can be led back to the same notation. On the other hand, only SOM looks at the structure of a business process.

2 Petri Nets as a Basic Method for Business Process Modelling

Petri nets (Reisig 2010) are a basic notation for modelling information flows of systems. They consist of two types of components, places and transitions. Transitions can have input places as well as output places. In the simplest case, a transition can fire, if all its input places are at least marked with one token, causing all its output places to get an additional token. Thus, a petri net models only the behaviour of a system, its structure is ignored. The components of the systems, e. g. company and customer, cannot be shown.

Of course there are different ways to model a business process as a petri net, depending on

* Corresponding author.

E-mail. elmar.sinz@uni-bamberg.de

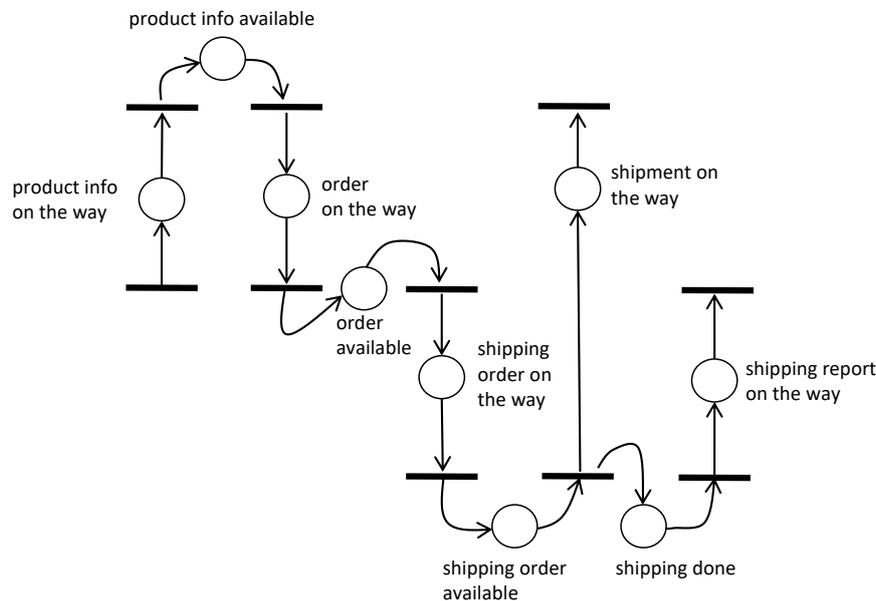


Figure 1: The Sample Business Process as a Petri Net

the goals of the model. For example, the transfer of information or goods can be modelled using separate places, or can be suppressed. In the following, for reasons of comparability of the different business process modelling methods, the transfer is modelled.

Figure 1 shows the resulting business process as a petri net. It starts on the left side with the product information, followed by an order if the product information conforms to the requirements. The order is succeeded by a shipping order, causing the shipment and the generation of a shipment report.

3 Event-Driven Process Chains (EPC) as a Method for Business Process Modelling

Event-driven process chains (EPC) (Nüttgens 2017) are a well-known method for business process modelling proposed in the ARIS approach (Scheer 1998). Figure 2 shows the business process of Figure 1 represented as EPC. To facilitate the comparison of the two methods, conforming components are arranged similarly. The concepts are bridged as follows:

- Events (represented as hexagons) correspond one-to-one to the places of petri nets.
- Transitions are replaced by functions. While a transition is not time-consuming, a function may be.
- Connectors (not used in Figure 2) are represented by circles labelled with the Boolean operators AND, OR and XOR. They can be combined and used to specify pre- and post-conditions of functions. Compared to petri nets, connectors are an additional feature, leading to an extension of semantics. In a petri net, the equivalent of a connector has to be modelled by additional transitions and places, considering that a transition can fire if all preceding places are at least marked once (AND) or all preceding transitions feed a common place which is the single input for the particular transition (OR).

A remarkable difference between a petri net and an EPC is that an EPC always starts and ends with an event (Scheer and Thomas 2005). Thus, in the current example three additional events have to be included (in Figure 2 grey shaded).

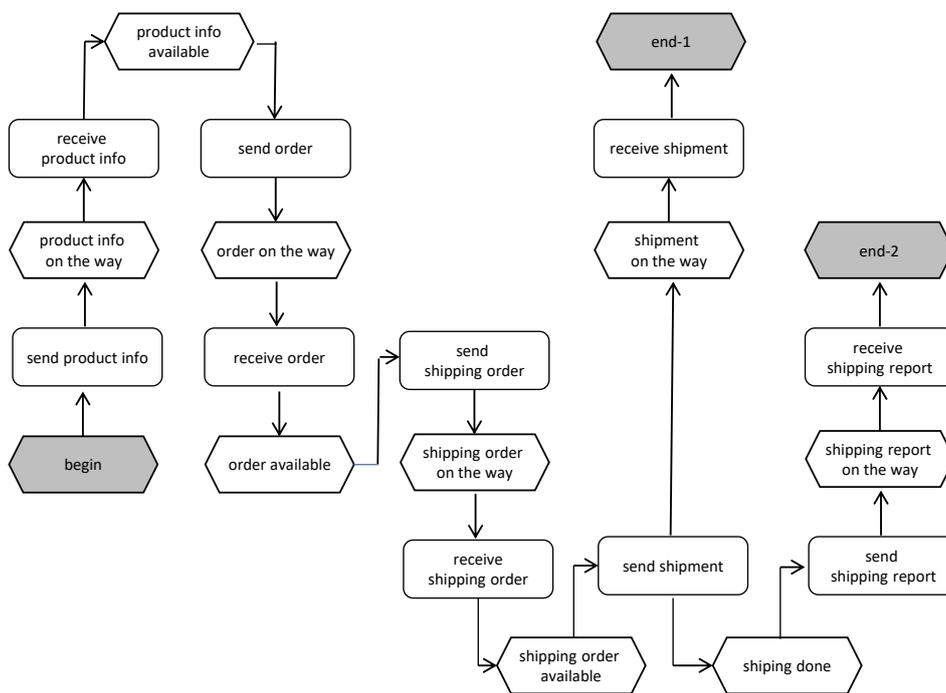


Figure 2: The Sample Business Process as an Event-Driven Process Chain

Functions can be related with e.g. information objects and organizational units. These are referencing items only, they do not constitute an own model within the context of an EPC.

4 Business Process Model and Notation (BPMN) as a Method for Workflow Modelling

Business Process Model and Notation (BPMN) is a popular method for workflow modelling (OMG 2017; Weske 2012). The term workflow instead of business process means that a workflow specifies the activities and relations between activities while executing one or more business tasks (Pütz and Sinz 2010). By contrast, a business process describes business tasks and event relations between tasks. It is strictly goal-oriented. Workflow models focus as well as business process models on the behaviour of a system.

The corresponding concepts between a petri net and a BPMN schema are bridged as follows (Figure 3):

- Transitions correspond to activities.

- Places occur as start events, as end events, or as events connecting activities. The latter can be omitted due to the fact, that a BPMN schema models a workflow. Here, events inside the execution of a business task are not in the foreground.
- Gateways (not used in Figure 3) allow e.g. the parallel split or the merging of flows.

The most interesting concepts are pools. Pools are participants, shown as rectangles which surround activities. The rectangles are labelled with the names of the participants, who execute the corresponding activities. Inside a pool, activities are related by sequence flows (solid lines); between pools there are message flows (dashed lines).

Another semantic detail can be traced back to the fact that a pool is a participant. Compared to a petri net, the control flow of activities executed by one participant is continuous. For example, the sequence flow between "send product info" and "receive order" cannot be found in the petri net (Figure 1) and by the way neither in the EPC (Figure 2).

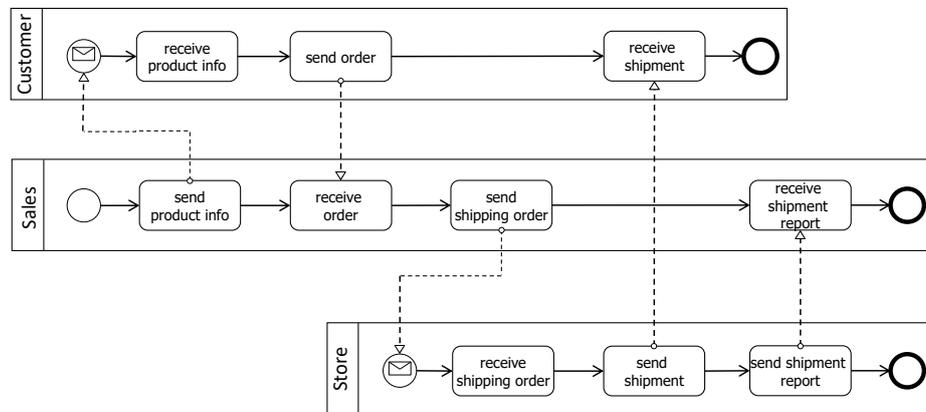


Figure 3: The Sample Workflow as a BPMN Schema

5 Semantic Object Model (SOM) as a Method for Business Process Modelling

In the SOM method (Ferstl and Sinz 1995; Ferstl and Sinz 2005; Ferstl and Sinz 2013, p. 194) the behaviour of a business process is modelled by a task-event schema. The following issues bridge between the concepts of a petri net and a task-event schema (Figure 4):

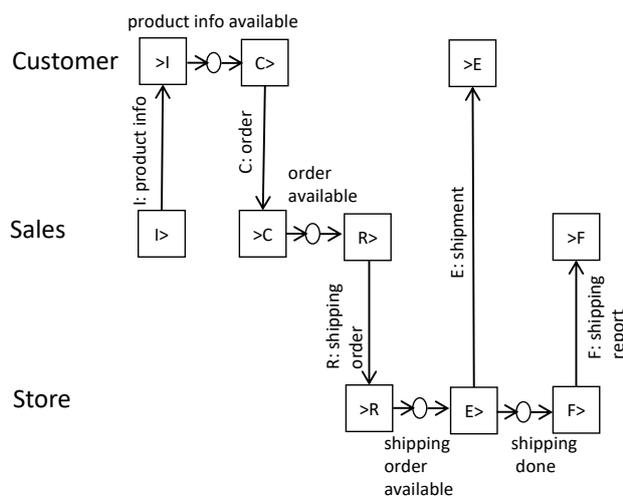


Figure 4: The Sample Business Process as a SOM Task-Event Schema (Behaviour)

- **Tasks:** Transitions are replaced by tasks. A task is a goal-oriented operation on a task object, released by and producing events.
- **Object-orientation:** Tasks are combined in an object-oriented way. All tasks operating on the same task object form an object. In Figure 4 e. g. the object customer contains the tasks >I (receive a product info), >C (send an order), and >E (receive a shipment).
- **Places** correspond to internal events of an object, i.e. an event connecting two tasks of one object.
- **Transactions:** An event from an object to another is represented as a transaction. A transaction causes a synchronized execution of the two tasks, e.g. >C (send an order) and >C (receive an order) must be completed in one transaction. >C as well as >C cannot terminate separately. Therefore, the event of a transaction is not displayed in the task-event schema.
- **Pre- and post-conditions:** Tasks can be complemented by pre- and post-conditions. These are Boolean expressions, e.g. if shipping is done only once in the afternoon, task E> could have the pre-condition "shipping order available AND time = 5 p.m."
- **Colored petri nets:** the tokens of an event can be distinguished and therefore assigned to an instance of a task operation.

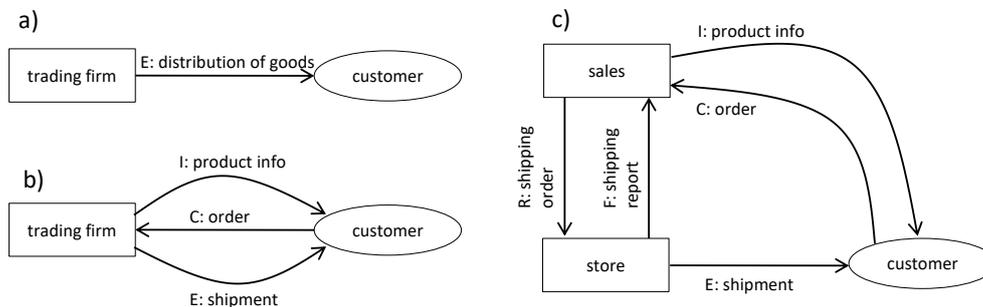


Figure 5: The Sample Business Process as alternate SOM Interaction Schemas (Structure)

Besides the task-event schema as the behaviour view on a business process, SOM provides an interaction schema as the structure view. Structure and behaviour view are adjusted, but an advantageous modelling always starts with the structure view. The structure view shows the decomposition of a system or a business process respectively, thus revealing sub-transactions and sub-objects (Figure 5).

A SOM interaction schema always starts with the most aggregated system view on the universe of discourse (trading firm) and its environment (customer). The object denoting the universe of discourse is connected with each environment object by one transaction (distribution of goods) (Fig 5a). Now the transaction(s) or the universe of discourse have to be further decomposed. It's a good idea to continue with the transaction, which is deconstructed according to the rule

$$T(O, O') ::=$$

$$[[T_i(O, O') \text{ seq}] T_c(O', O) \text{ seq}] T_e(O, O').$$

This means: replace the transaction on the left side ($T(O, O')$) by an initiating transaction ($T_i(O, O')$) sequentially followed by a contracting transaction ($T_c(O', O)$) sequentially followed by an enforcing transaction (T_e) (Fig. 5b). Initiating transaction as well as initiating and contraction transaction can be omitted. The rule is called the negotiation principle and is one of the two fundamental coordination principles the SOM model supports. The other coordination principle

is the feedback control principle. It is given by the rule

$$O ::= \{ O', O'', T_r(O', O''), [T_f(O'', O')] \}.$$

An object O is replaced by the set of objects O' and O'' , a control transaction $T_r(O', O'')$ from O' to O'' and a feedback transaction $T_f(O'', O')$ from O'' to O' . The latter can be omitted if there is only a controlled system. As shown in Figure 5c, the object trading firm is decomposed into the sub-objects sales and store as well as the control transaction shipping order and the feedback transaction shipping report. The sub-transactions from the first decomposition are linked to the new sub-objects.

Given an appropriate software tool (Ferstl et al. 2016), one can slide up and down the object decomposition as well as the transaction decomposition, each level showing a consistent decomposition of the system and associated with a corresponding task-event schema.

6 Conclusion

As pointed out, EPC, BPMN and the task-event schema of SOM can be explained in terms of petri nets. The modelling methods amend the petri net semantics differently to specify the behaviour of a business process or a workflow respectively. For example, EPC denote every event and are easy to read, BPMN show participants and their communication using message flows, and SOM points out the synchronous execution of a transaction between different objects.

Only SOM has a structure model, having a lot of benefits. The structure shows the decomposition of a system revealing sub-objects and sub-transactions. The reason for having a structure model is the object-orientation of SOM. The combination of "objects having tasks" and "two tasks of different objects are driving a transaction" is the prerequisite for the decomposition of a model.

The opportunity to zoom in and out the system and having a consistent model on each level adds a "third dimension" to business process modelling. On each level of aggregation a behaviour model can be assigned. The structure model is among others the platform for model driven architecture (e. g. Pütz and Sinz 2010).

BPMN could be amended with a structure model when giving up the semantics of "a pool is a participant" and replacing it by "a pool is an object". Without investigating all the details, this would be a great step forward.

References

- Ferstl O. K., Sinz E. J. (1993) Geschäftsprozeßmodellierung. In: WIRTSCHAFTSINFORMATIK 35(6), pp. 589–592
- Ferstl O. K., Sinz E. J. (1995) Der Ansatz des Semantischen Objektmodells (SOM) zur Modellierung von Geschäftsprozessen. In: WIRTSCHAFTSINFORMATIK 37(3), pp. 209–220
- Ferstl O. K., Sinz E. J. (2005) Modeling of Business Systems Using SOM. In: Bernus P., Mertins K., Schmidt G. (eds.) Handbook on Architectures of Information Systems. International Handbook on Information Systems. Springer, Berlin, pp. 347–367
- Ferstl O. K., Sinz E. J. (2013) Grundlagen der Wirtschaftsinformatik, 7th ed. Oldenbourg, München
- Ferstl O. K., Sinz E. J., Bork D. (2016) Tool Support for the Semantic Object Model. In: Karagiannis D., Mayr H. C., Mylopoulos J. (eds.) Domain-Specific Conceptual Modeling. Springer, Berlin, pp. 291–310
- Nüttgens M. (2017) EPK. In: Kurbel K., Becker J., Gronau N., Sinz E., Suhl L. (eds.) Enzyklopädie der Wirtschaftsinformatik – Online-Lexikon 7th ed. Oldenbourg, München, 13.9.2013 <http://www.enzyklopaedie-der-wirtschaftsinformatik.de> Last Access: 25/09/2014
- OMG (2017) Business Process Model and Notation (BPMN). Version 2.0.2. 2014 <http://www.omg.org/spec/BPMN/2.0.2/> Last Access: 16/11/2017
- Pütz C., Sinz E. J. (2010) Model-driven Derivation of BPMN Workflow Schemata from SOM Business Process Models. In: Enterprise Modelling and Information Systems Architectures 5(2), pp. 57–72
- Reisig W. (2010) Petrinetze. Modellierungstechnik, Analysemethoden, Fallstudien. Vieweg + Teubner, Wiesbaden
- Scheer A.-W. (1998) ARIS – Modellierungsmethoden, Metamodelle, Anwendungen, 3rd ed. Springer, Berlin
- Scheer A.-W., Thomas O. (2005) Geschäftsprozessmodellierung mit der ereignisgesteuerten Prozesskette. In: Das Wirtschaftsstudium 34(8-9), pp. 1069–1078
- Vossen G., Becker J. (eds.) (1996) Geschäftsprozessmodellierung und Workflow-Management. Modelle, Methoden, Werkzeuge. Thomson, Bonn
- Weske M. (2012) Business Process Management. Concepts, Languages, Architectures, 2nd ed. Springer, Berlin

This work is licensed under a Creative Commons 'Attribution-ShareAlike 4.0 International' licence.

