Subject-Oriented Business Process Management

Albert Fleischmann, Werner Schmidt, and Christian Stary

Abstract Business is increasingly characterized by interactions among responsive stakeholders rather than the functional decomposition of work. The subject-oriented approach to BPM (S-BPM) is considering this requirement by sending and receiving messages enveloping functional task accomplishment. Subjects represent the information processing entities in a business process. They communicate with each other in order to coordinate their work by exchanging information which is contained in so called business objects.

Subjects are embedded into some organizational and technical environment. Agents assigned to subjects (people or technical equipment) execute the actions defined in the subject specification. Business objects can be implemented as information containers or any tangible goods which are transported between agents. This separation of logical model and its implementation increases the flexibility of business processes management, as revealed by several academic and industrial S-BPM projects. Finally, the structuring of processes models as interacting entities facilitates coordinating business process management activities.

1 Introduction

The hidden paradigm behind modeling business processes in an enterprise is based on Ford's and Taylor's idea of sequencing activities and taking the best in class approaches (Taylor 1911; Ford and Crowther 1922). It has once proven to be suitable for the mass production of goods. While the paradigm is still the basic design assumption for shaping business processes, the environmental and social basis for enterprises has changed significantly. Business has moved from mass good production to massive personalized services and goods, where customers can place

A. Fleischmann (✉)
Metasonic AG, Hettenshausen, Germany
e-mail: Albert.Fleischmann@metasonic.de

unpredicted change requests nearly at any time. The fact that such events occur unpredicted does not mean they occur rarely. Research on principles of good BPM has therefore identified the context-awareness (as the ability to be sensitive towards the nature of different business areas) as a key principle for BPM (vom Brocke et al. 2014). In volatile environments, as given in the current situation of the economy, the exception to the 'ideal path' is likely to become basically the routine. How the reaction to such unpredicted events looks like is shown by the inflationary usage of emails, instant messages, phone calls and meetings. It seems that communication is about to become the new paradigm (Fleischmann et al. 2013d).

Putting massively personalized services on top of complex products asks for new architectural structures. Ford's hidden paradigm fails to master the resulting architectural complexity due to the lack of the concept of "communication". Rather, Luhmann's sociological understanding of systems as communicating entities has the potential to become the novel perspective on business operation (Berghaus 2011). Luhmann considers an organization as comprised of communication, the smallest organization being the communication between two information processing entities. Such a perspective is grounded on abstracting from concrete actors in the first step, while putting them into their particular context when detailing communication acts. Hence, it allows a stakeholder perspective while preserving coherent organizational behavior (systemic perspective).

Subject-oriented business process management (S-BPM) follows this kind of communication-oriented paradigm: Each functional step in task accomplishment is framed by communication acts including relevant business objects. This concept allows overcoming several deficiencies of traditional, activity-based Business Process Management (BPM) approaches in a business world which is increasingly characterized by unpredictable events.

The S-BPM approach has been evaluated in several industrial projects and application domains, among them:

- IT-service management: FI-TS, an IT service provider with around 500 employees in the banking area, has specified its service order and delivery process in S-BPM. In order to implement that process they have used an existing tool suite (Konjack 2010). With the solution automatically generated from the process specification several hundred process instances are handled a month. FI-TS estimates that they could reduce their execution time by more than 20 %.
- Customer knowledge management: NEC has developed a detailed methodology for BPM based on S-BPM which allows managing the development and maintenance of very complex processes in large organizations (Nakamura et al. 2011). NEC estimates that could reduce their execution time for several processes by more than 70 %.
- Incident management: The Swiss Telecom has implemented several processes with S-BPM. An incident management process is used by several partners of Swiss Telecom. In Walke et al. (2013) details can be found about an iPhone order process. This solution allows customers to order on their own and without the assistance of a contact center agent of Swisscoms hotline. The process was modeled with a S-BPM modeling tool, out of that modell a workflow is

generated automatically, access to various data bases is added manually and some adaptions were made to the user interface. For monitoring the data recorded by the workflow engine are used and to create some reports ARIS PPM is used. Some much more details can be found on the presentation recorded at the S-BPM ONE 2013 which is available on Youtube (see Walke et al. 2013).

Beyond that, S-BPM is also in the production and evaluation phase at several German car manufacturers,[1] and service providers, among them TILAK, an Austrian health care provider. In the latter case, S-BPM has been embedded prototypically into a systemic approach to organizational development (Augl 2012). When transforming existing patterns of communication to contextual collaboration different professionals within (health) expert organizations need to negotiate and agree on interactions empowering the organization for high-quality patient care.

Essentially, S-BPM subjects serve as "boundary objects" for the coordination, translation and creation of shared meaning. In this way, models become accessible to discussion, validation, negotiation, and change, finally, shortening traditional BPM life-cycles and leading to an open BPM life cycle (Fleischmann et al. 2013c). However, the active participation of actors needs to be ensured. In the Austrian health care project, members of a special care unit managed to develop novel communication patterns for daily scheduling of physicians. The newly established processes contribute significantly to increased performance of the special clinic and the entire health care organization. Although the complexity has been enriched enlarging the scope of planning (now including teaching duties of staff), timely access to relevant data has been established and overall transparency of the planning process has be increased. In the following sections we introduce the S-BPM approach along some fundamental business requirements. In Sect. 2 we motivate the paradigm change from activity/function-oriented approaches to communication-oriented ones, as it enables a more flexible approach to modeling, and thus to BPM. In Sect. 3 we focus on the core activity in BPM, namely targeted modeling. S-BPM supports starting either from scratch (like in function-oriented approaches), or from a generic multi-party scheme by restricting general behavior sequences to task-specific behavior specifications. The latter bridges the gap to executing business process models, as the generic scheme provides complete, i.e. ready-to-execute behavior models. Executable subjects represent agents encapsulating subject behavior. Consequently, in Sect. 4, we address architectural implementation issues. S-BPM decouples modeling from organizational and technical implementation while preserving a coherent process execution scheme. In the second part of this section we tackle handling instances of business models in concrete organizational settings. Both, decoupling organizational and technical implementation from modeling, demonstrate the effectiveness and efficiency of S-BPM implementations, and lead to a clarification of roles in BPM (cf. Fleischmann et al. 2012a). In Sect. 5 we conclude the chapter by wrapping up the introduced concepts, and sketching current and future research activities.

---

[1] Because of company internal reasons it is not possible to mention the companies, the projects and results.

2 Business as Dynamic Collaborative Communication Processes

In this section we lay ground for S-BPM by discussing the cooperation and communication pattern between the involved parties and corresponding IT and communication systems. In addition, we address agility with respect to behavior – processes need to be defined in a flexible way to give people executing these processes some autonomy in their decision making. We elaborate the concept of S-BPM and its features using an example illustrating the most important capabilities. Further features like multi-subjects, process networks etc. are described in (Fleischmann et.al. 2012b).

2.1 *Subject-Driven Business Processes*

Subjects represent the behavior of an active entity. A specification of a subject does not say anything about the technology used to execute the described behavior. This is different to other encapsulation approaches, such as multi-agent systems.

Subjects communicate with each other by exchanging messages. Messages have a name and a payload. The name should express the meaning of a message informally and the payloads are the data (business objects) transported. Internally, subjects execute local activities such as calculating a price, storing an address etc.

A subject sends messages to other subjects, expects messages from other subjects, and executes internal actions. All these activities are done in sequences which are defined in a subject's behavior specification.

Subject-oriented process specifications are embedded in a context. A context is defined by the business organization and the technology by which a business process is executed.

Subject-oriented system development integrates established theories and concepts. It has been inspired by various process algebras (see e.g. Hoare 1985; Milner 1989, 1999), by the basic structure of nearly all natural languages (Subject, Predicate, Object) and the systemic sociology developed by Niklas Luhmann (an introduction can be found in Berghaus 2011). According to the organizational theory developed by Luhmann the smallest organization consists of communication executed between at least two information processing entities (Berghaus 2011). The integrated concepts have been enhanced and adapted to organizational stakeholder requirements, such as providing a simple graphical notation, as detailed in the following sections.

2.2   Subject Interaction and Behavior

We introduce the basic concepts of process modeling in S-BPM using a simple order process. A customer sends an order to the order handling department of a supplier. He is going to receive an order confirmation and the ordered product by the shipment company. Figure 1 shows the communication structure of that process. The involved subjects and the messages they exchange can easily be grasped.

Each subject has a so-called input pool which is its mail box for receiving messages. This input pool can be structured according to the business requirements at hand. The modeler can define how many messages of which type and/or from which sender can be deposited and what the reaction is if these restrictions are violated. This means the synchronization through message exchange can be specified for each subject individually.

Messages have an intuitive meaning expressed by their name. A formal semantic is given by their use and the data which are transported with a message. This means in S-BPM there is only the concept message in contrary to speech act theory. In speech act theory messages can have a basic semantic like request, response etc. For instance, the FIPA-ACL Communicative Act library consists of 22 communication acts or performatives (e.g. see Billifemine et al. (2007), p. 19). Moreover, these communication acts can be emulated by the basic messages used in S-BPM. In a layer below the communication structure, according to the interaction behavior of each subject, is described. Figure 2 depicts the behavior of the subjects "customer" and "order handling".

In the first state of its behavior the subject "customer" executes the internal function "Prepare order". When this function is finished the transition "order prepared" follows. In the succeeding state "send order" the message "order" is sent to the subject "order handling". After this message is sent (deposited in the input pool of subject "order handling"), the subject "Customer" goes into the state "wait for confirmation". If this message is not in the input pool the subject stops its execution, until the corresponding message arrives in the input pool. On arrival the subject removes the message from the input pool and follows the transition into state "Wait for product" and so on.

The subject "Order Handling" waits for the message "order" from the subject "customer". If this message is in the input pool it is removed and the succeeding function "check order" is executed and so on.

The behavior of each subject describes in which order it sends messages, expects (receives) messages and performs internal functions. Messages transport data from the sending to the receiving subject, and internal functions operate on internal data of a subject. These data aspects of a subject are described in Sect. 2.3. In a dynamic and fast changing world, processes need to be able to capture known but unpredictable events. In our example let us assume that a customer can change an order. This means the subject "customer" may send the message "Change order" at any time. Figure 3 shows the corresponding communication structure, which now contains the message "change order".
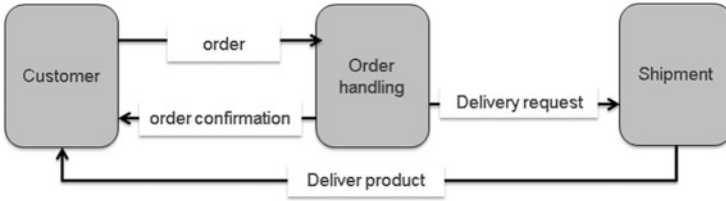
Fig. 1 The communication structure in the order process

Due to this unpredictable event the behavior of the involved subjects needs also to be adapted. Figure 4 illustrates the respective behavior of the customer.

The subject "customer" may have the idea to change its order in the state "wait for confirmation" or in the state "wait for product". The flags in these states indicate that there is a so-called behavior extension described by a so-called nondeterministic event guard (Fleischmann et al. 2013a, d). The non-deterministic event created in the subject is the idea "change order". If this idea comes up, the current states, either "wait for confirmation" or "wait for product", are left, and the subject "customer" jumps into state "change order" in the guard behavior. In this state the message "change order" is sent and the subject waits in state "wait for reaction". In this state the answer can either be "order change accepted" or "order change rejected". Independently of the received message the subject "customer" moves to the state "wait for product". The message "order change accepted" is considered as confirmation, if a confirmation has not arrived yet (state "wait for confirmation").
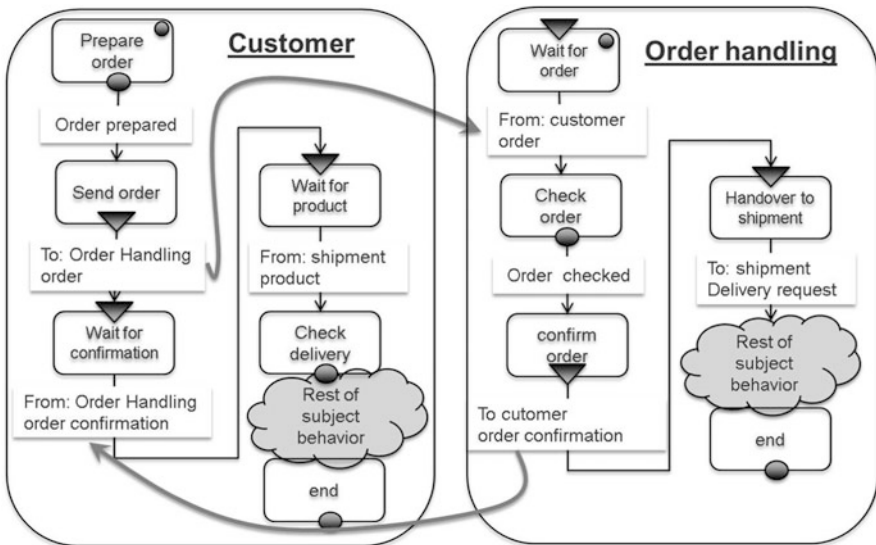

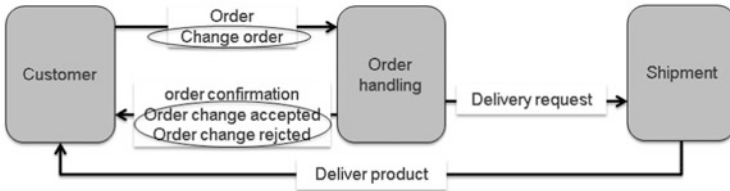**Fig. 2** The behavior of subjects

Fig. 3 The communication structure with change message

If the change is rejected the customer has to wait for the product(s) he/she has ordered originally.

Similar to the behavior of the subject "customer" the behavior of the subject "order handling" has to be adapted.

We have only captured the basic elements of S-BPM in this section. In order to model complex process systems, processes can be connected with each other in order to build networks. Describing these networks is a straightforward task, since the message mechanism as explained above can be used on the network layer, too. A precise and complete definition of the semantics of all S-BPM modeling elements can be found in the attachment of (Fleischmann et al. 2012a). The complete formal semantic specification as an abstract state machine (Börger and Stärk 2003) has only 9 pages. Due to this precise and formal specification, S-BPM models can be automatically converted in executable code (see Sect. 4).
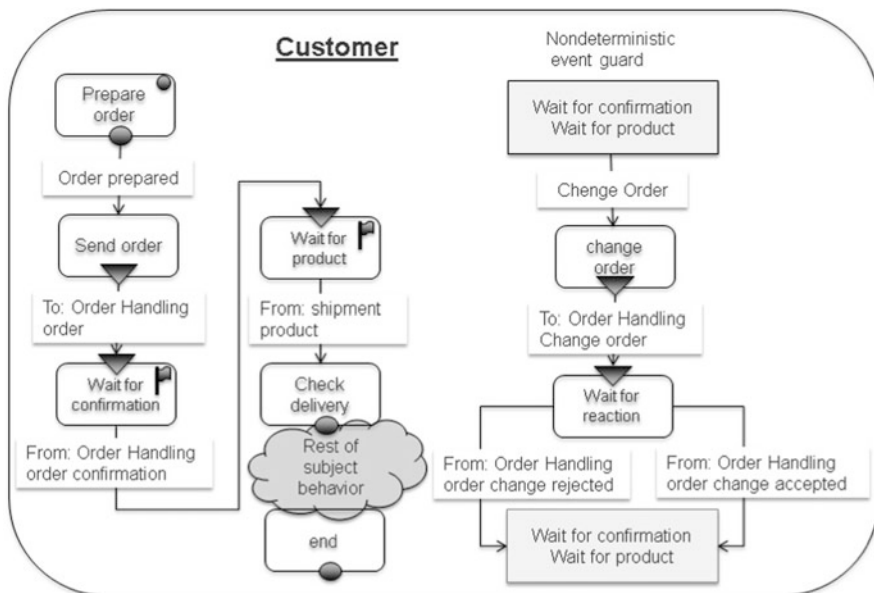

Fig. 4 Customer is allowed to change orders

2.3   Subjects and Objects

Up to now we did not mention data or the objects with their predicates, in order to get complete sentences comprising subject, predicate, and object. Figure 5 displays how subjects and objects are connected. The internal function "prepare order" uses internal data to prepare the data for the order message. This order data is sent as payload of the message "order".

The internal functions in a subject can be realized as methods of an object or functions implemented in a service, if a service-oriented architecture is available. These objects have an additional method for each message. If a message is sent, the method allows receiving data values sent with the message, and if a message is received the corresponding method is used to store the received data in the object (Fleischmann et al. 2013d). This means either subjects are the entities which use synchronous services as implementation of functions or asynchronous services are implemented through subjects or even through complex processes consisting of several subjects. Consequently, the concept Service Oriented Architecture (SOA) is complementary to S-BPM: Subjects are the entities which use the services offered by SOAs (cf. Sneed et al. 2012).
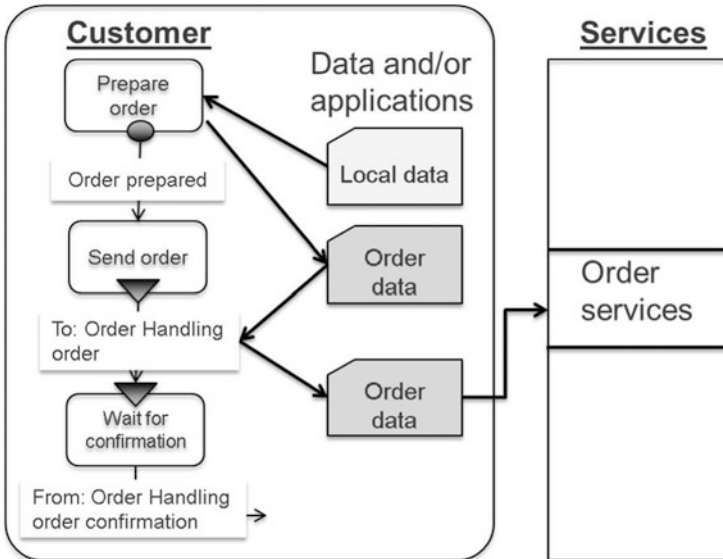

Fig. 5 Subjects and objects

3 Targeted Modeling Through Natural Language and 5-Symbol Articulation

This section motivates and details the use of standard-sentence semantics for the representation of business processes, either starting from scratch (constructing models), or from generic interaction patterns (restricting interactions according the organization of work). S-BPM originates also from the observation that humans, when structuring and describing their observed reality, use subjects, predicates, and objects. Each of them can be mapped to natural language entities. As already indicated above, the subject represents the active element, the predicate the action and the object is the entity on which the action is executed. Natural language supports human communication effectively, both in written and oral form. As humans use natural language structures as primary means to ensure mutual understanding (Börger and Stärk 2003), model descriptions for formal modeling could make use of it, in order to facilitate understanding models. In order to ensure coherence of specifications, the exchange of messages determines the flow of control (in contrast to function-oriented approaches).

The S-BPM modeling language captures the above mentioned constituent elements of natural language sentences. Models describe structural properties and behavioral alternatives, including the interaction occurring in the technical and/or organizational environment. S-BPM models can be transformed step by step into an executable application in a seamless way.

Modeling means to represent parts of the observed reality in terms of languages. In case of S-BPM natural language terms are used, as they allow for universal use and are familiar to stakeholders through daily communication. S-BPM uses the standard semantics for sentences, comprising subject, predicate and object:

- A subject is the starting point for describing a situation or events,
- activities denoted by predicates, whereas
- an object is the target of an activity (denoted by a predicate).

Existing modeling approaches tend to focus on predicates or objects, adding the subject for natural language explanations of the represented information (cf. identifying function trees before specifying eEPCs in ARIS (Scheer 2001)). For a more detailed discussion of S-BPM in the context of traditional approaches see Fleischmann et al. (2012a, p. 269).

Models address both, individual work tasks, and organizationally relevant ones. In the course of accomplishing their tasks, stakeholders receive work inputs and pass on results. Hence, interaction and communication, either direct or indirect, are to be considered as an essential activity for subject-oriented modeling.

Figure 6 contains the natural language description of a customer order process. It is the initial version of the order process we have also used above.

This simple order process can be modelled following two different approaches (Fleischmann et al. 2012b). They differ by the starting point of building a process specification. The traditional approach (modeling through construction) starts from

> *Order Process:*
>
> *A customer orders some goods. He/She sends an order to the purchaser. He/she is then going to receive an order confirmation after the order has been checked. Once the order is checked, the order is handed over to the shipment department where the delivery is prepared and executed.*

Fig. 6 Natural language description of a customer order process

the scratch ('empty sheet'), and the process model is constructed step by step. Task-relevant actors or systems need to be identified as the process specification evolves, and the lines of interaction need to be included as required for task accomplishment. The other approach (modeling through restriction) is only available in S-BPM. It starts with a generic process model which is restricted step by step. The generic process can be compared with the behaviour when each involved stakeholder uses e-mail: Each stakeholder can communicate with another stakeholder he or she is linked to. A process is derived from such a completely networked structure by removing communication lines step by step that are not relevant for business achievements. In the course of modelling the lines of interaction between subjects are adapted to those required for task accomplishment.

In the following both approaches will be explained in detail. In Sect. 3.1 the stepwise construction of a communication-based process model is detailed. In Sect. 3.2 the stepwise reduction of interaction between actors or acting components is explained. In either cases, actual or envisioned business processes need to be represented in a transparent and traceable way. Finally, in Sect. 3.3 we refer to tangible tools facilitating work knowledge elicitation and its subject-oriented representation.

3.1 Modeling by Construction

Subject-oriented modeling of processes applying the construction approach includes the following major activities:

- the subjects involved in a process,
- interactions they are part of,
- the messages they send or receive through each interaction, and
- the behavior of each subject encapsulating functions and interactions

In the following we detail them according to major modeling concerns.

3.1.1 Who Am I and Who Needs to Be Involved? Subjects and Their Interactions

As already mentioned subjects are abstract resources representing the parties involved in a process the modeling process might start with identifying the involved

subjects the messages they exchange. The result of that step is the Subject Interaction Diagram (SID) or communication diagram as it is already shown in Fig. 1.

After that step the behavior of each subject is defined.

3.1.2 How Do I Operate? Subject Behavior, States and State Transitions

Subject behavior is described by three states (send, receive, internal function) and transitions between these states. These states represent predicates (operations), which means, that they are active elements of the subject description. Services are being used to implement the states and state transitions necessary to exchange and manipulate business objects. When specifying the behavior of each subject, as shown in Fig. 3 for the customer and order handling, a sequence of sending and receiving messages, and activities to be set for task accomplishment need to be represented.

3.1.3 Which Objects Do I Have to Manipulate? Services and Business Objects

The description of a subject defines the sequence of sending and receiving messages, or the processing of internal functions, respectively. In this way, a subject specification contains the sequence of predicates. Predicates can be of the type "send", "receive" or "internal function", the latter dealing with specific objects, such as required when a customer orders some products. As a consequence at least one operation needs to be assigned to each state. Detailing the operations is not necessary at the modeling stage. It is a matter of an abstract object specification or of the integration of an existing application. As an example the operation could be represented by a transaction of an ERP system related to the regarded object, for instance the update of an order master data record. Figure 5 shows how the predicates of a subject are defined by means of objects.

As we abstract from implementation details in the course of modeling, it seems suitable to replace the term 'operation' by the more general term 'service'. A service is assigned to a state and thus triggered and processed if the state is reached. The name of the states and the names of the assigned services can be different as shown in Fig. 5 because in a state several services can be used in order to define the required functionality executed in a state. The end conditions correspond to links leaving the state. Each result link of a sending state is assigned to a named service. Before sending this a service is triggered to identify the content or parameters of a message. This service determines the values of the message parameters transferred by the message. Similarly, each output link of a receiving state is assigned to a named service. When accepting a message in this state that service is triggered to identify the parameters of the received message. The service determines the values of the parameters transferred by the message and provides them for further

processing. All services are triggered in a synchronous way, i.e. a subject only reaches its subsequent state once all services called in a certain state have been completed.

3.2 Modeling by Restriction

The restriction approach in S-BPM starts with an overall generic process model. This generic model represents some kind of chaotic process: Everybody communicates with everybody whenever he or she wants. The first modeling task is therefore to restrict the number of participants. This means modelers have to decide how many subjects are involved in the process to be described. In a scenario everybody is communicating with everybody the behavior of the involved subjects is identical. However, starting with generic process templates that are only defined by the number of involved parties a process can become more concrete step by step. The procedure requires several restriction steps:

1. Specify a generic template according to the number of parties involved in handling a certain business case (cf. Fig. 8)
2. Name the subjects accordingly
3. Remove message connections between subjects which are not necessary
4. Name messages and introduce message types accordingly
5. Adapt specification to actual subject behavior
6. Refine the structure of the business objects transmitted by the various messages

   In the following subsection these steps are exemplified.

3.2.1 Who Needs to Be Involved? Generic Process Model

Figure 7 shows a generic subject-oriented process model with three involved parties. It fits to the number of subjects we expect for the customer order process. This means a modeler needs to identify the number of subjects in a process initially. This is the only information he/she needs for the first step. Each of the parties exchange messages with another party. We want to show how this generic process is restricted step by step in order to get a process specification for the customer order process as described in Figs. 7 and 8.

Each subject can send messages with the name "Message" to any other subject any time. Figure 8 shows the behavior of the subject with the name "Subject1".

In the select state a subject decides whether it wants to send or to receive a message. To start a workflow it does not make sense to receive a message because all the other subjects are waiting for messages. This means the start subject will start with sending messages and the message exchange can begin. Choosing the send transition the subject goes into the state "prepare message and select address" and fills out the business object that is transmitted by the message "message" (see end of
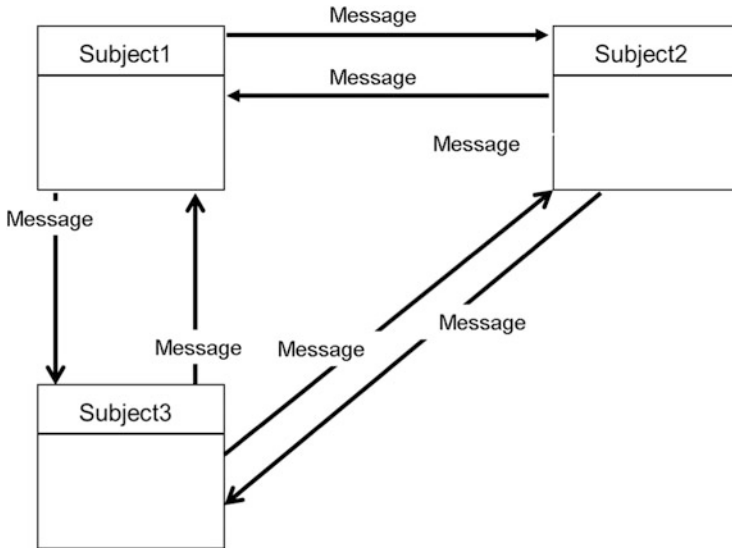
Fig. 7 Subject-oriented representation scheme for a 3-party-process

this section). After that the subject decides to which other subject the message with the business object as content will be sent.

In the select state a subject can also decide whether it wants to receive a message.[2] If there is a message for the subject available it can be accepted and a follow up action can be executed. It is not specified what the follow up action is. This is like receiving an e-mail. The receiver can interpret the content of an e-mail and knows what the corresponding follow up action is. The abort transitions back to the select state enable to step back in case a subject has made the improper choice.

The representation scheme can be easily created for any number of participants, following the same principles as shown for 3 parties. The behavior of each subject has to be adapted to the number of subjects in a process. In the send area transitions are required to send a message to every single new subject, and the same is necessary for the receive area. With that extension scheme the behavior for each type of multi-party process can be generated automatically.

Utilizing the message "Message" a business object is sent. The structure of this business object corresponds to the structure of a traditional e-mail with extensions like subject (attention: here the word "subject" has a different meaning. It can mean topic, issue, theme etc.), keywords and signature. Figure 9 depicts the specification of the business object "Message" in an XSD notation (XML Schema Definition).

---

[2] This choice can make sense for a start subject, from the second time on it goes into the select state.
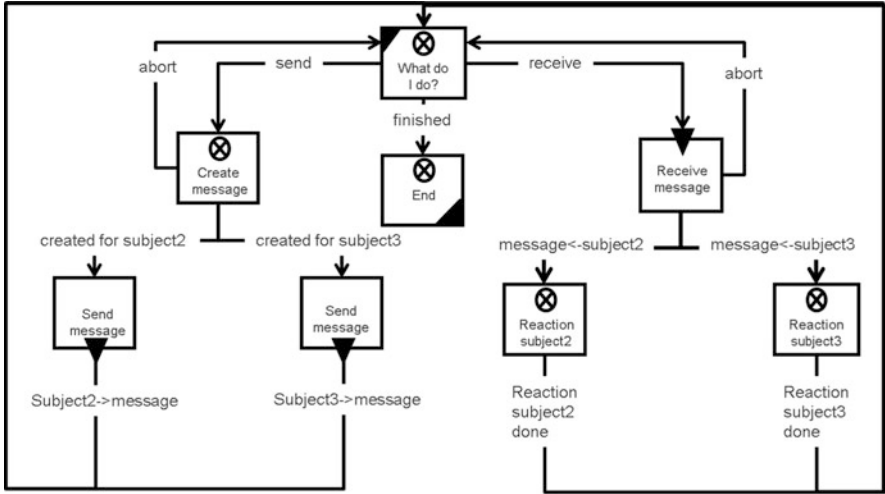
Fig. 8 Generic behavior of the subject "subject1"

3.2.2 How Do the Stakeholders Need to Interact? Adaption of Generic Scheme

Following the restriction steps in Sect. 3.2 a process specification is developed corresponding to the business requirements. In our example these steps result in a communication structure as shown in Fig. 10 and a behavior specification of the subject "customer" as shown in Fig. 11.

A comparison of Figs. 11 with 2 shows that modeling by restriction and construction does not necessarily result in identical models. Nevertheless both models need to deliver the requested business results.

With each restriction step the guidance for the subject holders is becoming more stringent to their actual task accomplishment. In this way, a subject-oriented system specification can guide the parties in a process for organizational development.
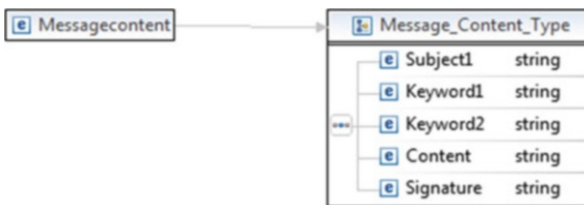

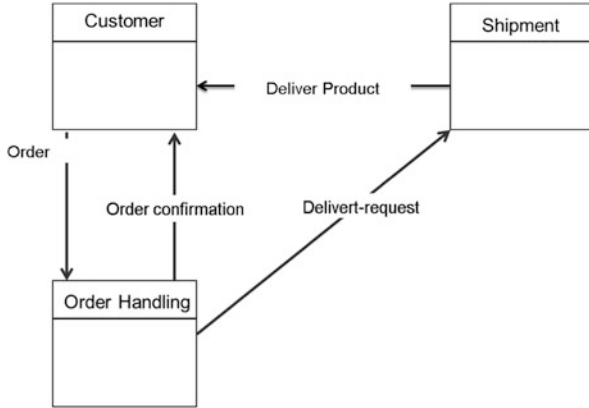**Fig. 9** Generic structure of the e-mail business object

Fig. 10 Subjects and exchanged messages

3.3 Tangible Modeling Support

S-BPM modeling is currently supported by various tools. Besides traditional computer-based 2D-modelling tools there exist modelling tools with tangible interfaces. These interfaces have been developed for supporting people who are not familiar with process modelling to capture their process knowledge. The tools help people to participate in creating models of those processes they work in, without forcing them to learn how to handle complex modelling tools (Fleischmann et al. 2013c, 2014a) – see Fig. 12.
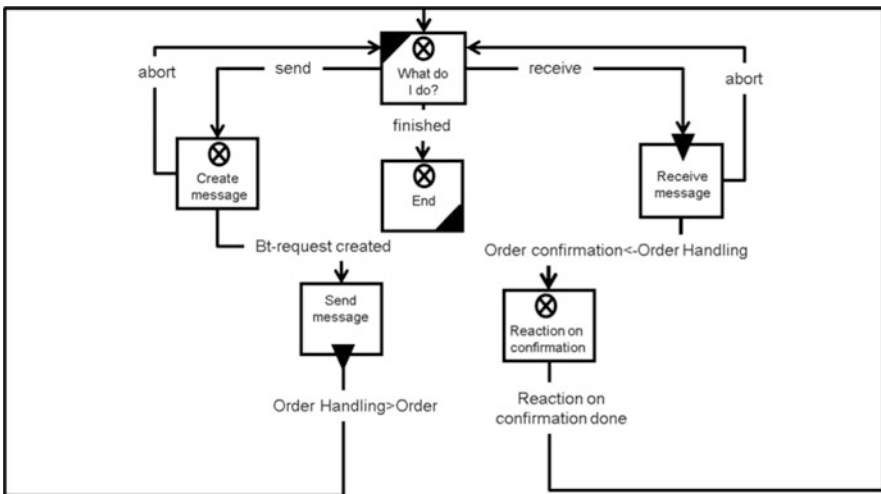

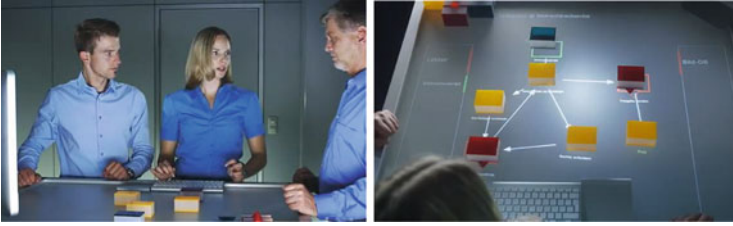**Fig. 11** Instantiated behavior of the subject "customer"

Fig. 12 Metasonic touch: tangible modeling support

Metasonic touch is a table on which the behavior of a subject can be described. Each activity type (Receive, Send, Do) is represented by a block with a different colour: Red for send, green for receive and yellow for doing. The lines between the states are drawn that the states which should be connected short are brought in contact with each other. The model created on the table is directly stored in a PC and can later changed by a common modeling tool. People who are involved in a process and have to define their behavior can stay around a modeling table and cooperate in a natural way to describe their subject behavior. The table produces a very communicative work atmosphere.

The experience with the modelling table are motivating for further research with tangible interfaces. One prerequisite for tangible interfaces are a very restricted number of symbols because of the number of clear colours and forms.

4   Implementation

In this section we first detail the benefits of decoupling (subject-oriented) business process models from implementation details while designing their organizational embodiment (Sect. 4.1). Then we discuss the capability of S-BPM model representations to be executed after validation without further transformations (Sect. 4.2).

4.1   Architecture Definition: Subject, Roles, and Agents

A set of subjects compose a business process. As already shown subjects can execute three different types of actions: Sending messages to other subjects, receiving messages from other subjects, and performing local actions on business objects. Business objects are transported via messages from the sending subject to the receiving subject. Local actions executed on a business object, such as creating, deleting or changing the object, can be considered as method invocations as known from object-oriented software development (see Fig. 13).

Agents are entities which are capable to execute actions. Each agent can be involved in several processes, where the same agent can enact different subjects
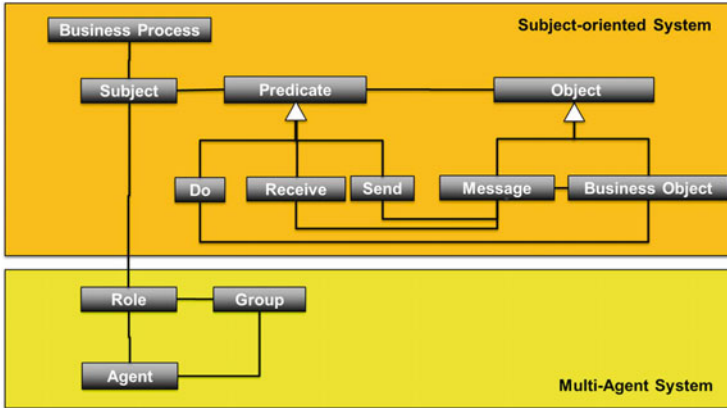
Fig. 13 Metamodel of S-BPM

across different processes. In turn, the same subject can be enacted by a single agent in one process or by a group of agents in another process. Roles are generalized combinations of subjects from different processes, cast into functional positions within the agent organization.

Roles are assigned to specific agents that execute the actions defined in subject descriptions. Agents can be people, software programs, robots etc. As a result, subjects may be enacted by heterogeneous groups consisting of different agent types. For instance, an "Order handling" subject may be enacted by a group of two interacting agents: software controlling the order handling workflow, and a human user entering required data.

Processes can be executed in different parts of an organization. The role "warehouse worker" acting in various processes exists in any subsidiary of a company. The role is the same but it is executed by different agents. A corresponding example is shown in Fig. 14.

In Fig. 14 two processes are shown: the ordering process, and a vacation application process. The role "Order handler" consists of the subjects "Order Handling" and "Employee", and the role "Warehouse worker" consists of the subjects "Employee" and "Shipment". The role "Order handler" is assigned to the group "Order Dept." and the role "Warehouse worker" to the group "Warehouse Dept.". The agents "Florian" and "Katrin" enact instances of the subjects "Order Handling" and "Employee", and the agents "Josef", "Christian" and "Thomas" enact instances of the subjects "Shipment" and "Employee".

The embedding of subjects into a socio-technical environment can be complex. As an example there might exist the rule that instances of the subject "shipment" has to be executed by agent "Josef" for a certain group of customers and by "Christian" for a different group etc. This means process models need to be embedded in their specific organizational setting, called process context or just context.
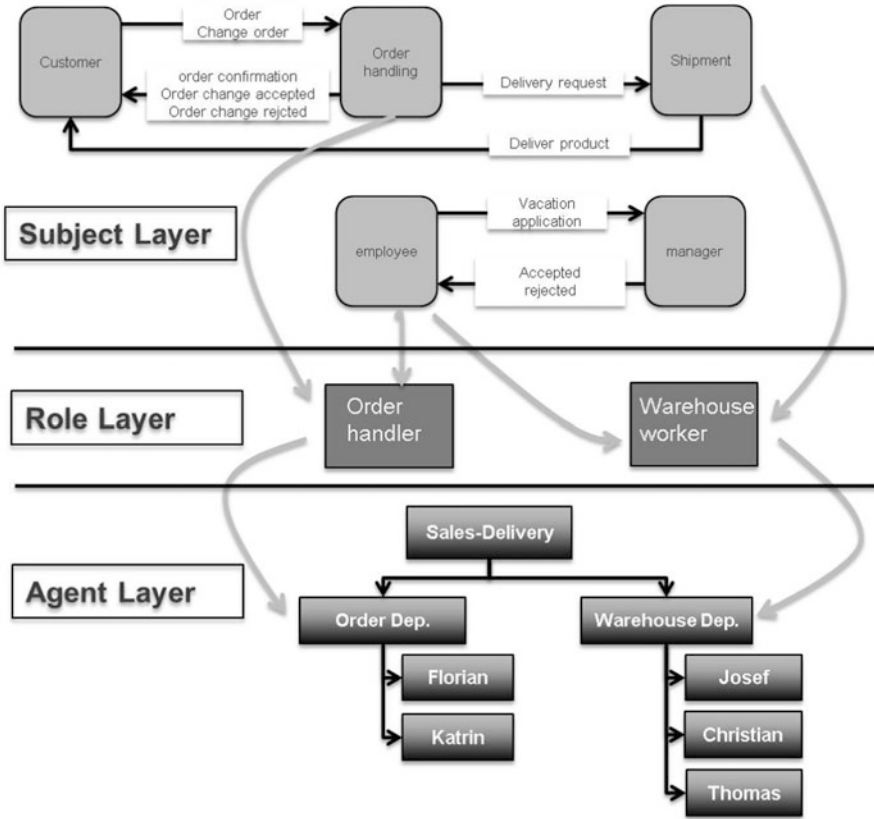
Fig. 14 Example of the subject, role, and agent relationship

A more detailed discussion about the relationship between subject-oriented processes and organizational aspects can be found in Fleischmann et al. (2012a, 2013e) and Lawall et al. (2013).

4.2 Automating Execution: Instances of Processes and Subjects

A process model including its embedding in its environment is only a pattern in order to properly react to certain business events. If the business event occurs, an instance of the process model in the corresponding environment is created in order to handle that business event. Such a business event can come from outside or inside of a process system. A business event from outside can be that a person wants to order some products. In order to handle that self-created event properly, an

instance of the corresponding process model is generated. This can be implemented by using a defined order number which is on each related document (in BPMN this id is called correlation key).

In S-BPM an instance is a complete, executable copy of the corresponding process model in the environment to which the business event belongs. An order arriving in subsidiary A causes the creation of a process instance in the environment to which subsidiary A belongs. An order arriving at subsidiary B causes the creation of an instance in its respective environment. Using that copy the business event is handled.

The creation of a process instance is not only caused by humans. It can also be caused by certain data states, a timer or by instances of other processes belonging to the same process network. Process instances can be created by subjects in other process instances if they send a message to a subject of a connected process and there is no corresponding instance which is related to the sending process instance. Then a corresponding process instance is created and linked with the initiating process.

5   Conclusion

Changes in society and business require different paradigms in business process management. Most of the current BPM approaches are still based on Taylorism and Fordism. In the Post-Fordism era job enrichment, self-management and communication have become central issues. Parties involved in processes want or need to organize their work by themselves. In global, highly distributed business operations known but still unpredictable events have become routine. Therefore communication-based BPM approaches have to recognize also spontaneous communication activities. Unpredictable activities like changing orders need also to be covered by suitable specification elements. In this chapter we have described such an approach.

S-BPM requires stakeholders to take responsibility for organizational developments by getting skilled in specifying business processes. This task should not be too challenging, as S-BPM models utilize natural language constructs (subject, predicate, object) and e-mail-like communication patterns between actors (subjects). In this way, individual members of an organization are enabled to contribute to coherent and intelligible process specifications. Moreover, resulting specifications can be processed without further transformation after validation.

The current state-of-the-art in S-BPM is just the point of departure for further developments:

• S-BPM lays ground for integrating social media communication in business procedures. As its common ground is the exchange of messages, informal relations between stakeholders need to be researched, in order to understand

business operations better and to implement sophisticated concepts, such as highly interactive customer knowledge management.

- Individualization of task accomplishment could be enforced on the basis of normative business operations, also termed standard operational procedures: S-BPM-models allow describing how to achieve a work result in a variety of ways in a coherent and consistent way. The actor/role/system-specific encapsulation of behavior in S-BPM is the key enabler for allowing diversity.
- Coopetitive behavior can be implemented on a process level. Competitors joining (ad-hoc) networks for innovative biddings or service provisions can keep their organizational assets encapsulated, offering only communication interfaces while hiding operational details as part of their USP.

Each of the above mentioned issues represents a research topic that should enlarge the scope of applying BPM, as even private, but societally relevant processes require stakeholder-specific communication and interaction.

References

Augl M (2012) Building a conceptual roadmap for systemic change – a novel approach to change management in expert organizations in health care. In: Stary C (ed) S-BPM ONE – scientific research, LNBIP, vol 104. Springer, Berlin/Heidelberg, pp 43–61

Berghaus M (2011) Luhmann leicht gemacht. Böhlau Verlag, Köln

Billifemine F, Caire G, Greenwood D (2007) Developing multi-agent systems with JADE. Wiley, Chichester/Hoboken

Börger E, Stärk R (2003) Abstract state machine – a method for high level system design and analysis. Springer, Berlin/Heidelberg

Fleischmann A, Schmidt W, Stary C, Obermeier S, Börger E (2012a) Subject-oriented business process management. Springer, Berlin/Heidelberg

Fleischmann A, Schmidt W, Stary C (2012b) A primer to subject-oriented business process modeling. In: Stary C (ed) S-BPM ONE – scientific research, LNBIP, vol 104. Springer, Berlin/Heidelberg, pp 218–239

Fleischmann A, Schmidt W, Stary C, Strecker F (2013a) Nondeterministic events in business processes. In: La Rosa M, Soffer P (eds) Business process management workshops, LNBIP, vol 132. Springer, Berlin/Heidelberg, pp 364–377

Fleischmann A, Schmidt W, Stary C (2013b) Open S-BPM = open innovation, 5th international conference, S-BPM ONE 2013, CCIS 360, Springer, Berlin/Heidelberg, pp 295–320

Fleischmann A, Schmidt W, Stary C (2013c) Subject-oriented BPM = socially executable BPM, workshop on Social Business Process Management (SBM 2013). In: Proceedings of the 15th IEEE conference on business informatics (CBI 2013). IEEE Computer Society Press, Vienna, pp 399–406

Fleischmann A, Schmidt W, Stary C (2013d) (Re-)Justifying BPM: a quest for the interaction turn reviewing subject-oriented BPM. In: Proceedings of the 15th IEEE conference on business informatics (CBI 2013). IEEE Computer Society Press, Vienna, pp 228–233

Fleischmann A, Kannengießer U, Schmidt W, Stary C (2013e) Subject-oriented modeling and execution of multi-agent business processes. In: Proceedings of the 2013 IEEE/WIC/ACM international conferences on web intelligence (WI) and intelligent agent technology (IAT). IEEE Computer Society Press, Atlanta, pp 138–145

Fleischmann A, Schmidt W, Stary C (2014a) Tangible or not tangible – a comparative study of interaction types for process modeling support. Proceedings of the 16th international conference on human-computer interaction (HCII 2014) in Heraklion

Ford H, Crowther S (1922) My life and work. Garden City Publishing Company, Inc., New York

Hoare A (1985) Communicating sequential processes. Prentice Hall, Englewood Cliffs

Konjack G (2010) Case study: AST order control processing. In: Buchwald H et al (eds) S-BPM ONE – setting the stage for subject oriented process management, CCIS, vol 85. Springer, Heidelberg/Berlin, pp 115–120

Lawall A, Schaller T, Reichelt D (2013) Integration of dynamic role resolution within the S-BPM approach. In: Fischer H, Schneeberger J (eds) S-BPM ONE – running processes, CCIS, vol 360. Springer, Heidelberg/Berlin, pp 21–33

Milner R (1989) Communication and concurrency. Prentice Hall, New York

Milner R (1999) Communicating and mobile systems: the Pi-calculus. Cambridge University Press, Cambridge

Nakamura S, Tan T et al (2011) CGAA/EES at NEC Corporation, Powered by S-BPM: the subject-oriented BPM development technique using top-down approach. In: Schmidt W (ed) - S-BPM ONE – learning by doing – doing by learning, CCIS, vol 213. Springer, Berlin/Heidelberg, pp 215–231

Scheer A-W (2001) ARIS – Modellierungsmethoden, Metamodelle, Anwendungen, 4th edn. Springer, Heidelberg/Berlin

Sneed HM, Schedl S, Sneed SH (2012) From legacy code to a business process model IEEE 6th international workshop on the Maintenance and Evolution of Service-Oriented and Cloud-Based Systems Riva del Garda, 2012

Taylor FW (1911) The principles of scientific management. Harper and Brothers, New York and London, LCCN 11010339, OCLC 233134

vom Brocke J, Schmiedel T, Recker J, Trkman P, Mertens W, Viaene S (2014) Ten principles of good business process management. Bus Proc Manage J (BPMJ) 20(4)

Walke T, Witschi M, Reiner M (2013) Case Study @ Swisscom (Schweiz) AG: iPhone 5 self-service order app and process-workflows. In: Fischer H, Schneeberger J (eds) S-BPM ONE – running processes, CCIS, vol 360. Springer, Heidelberg/Berlin, pp 264–273, A video about that articel with additional information can be found on Youtube under the link: http://www.youtube.com/watch?v=e3OvQsggtDg