Business Process Modeling based on UML Activity Diagrams

Michal Radecký *
michal.radecky@vsb.cz

Ivo Vondrák*
ivo.vondrak@vsb.cz

Abstract: The importance and role of the business processes and their modeling are still growing, nowadays. So, it is important to offer the tools and methods which will combine the facilities and possibilities of process modeling techniques on the one hand and requirements consequent on the demands or potentialities of "non-informatics process's user or business user" on the other hand. This paper describes the possible solution of this integration. It is approach to the modeling of business processes by a graphical language that is based on the UML Activity Diagrams. This extended version of Activity Diagrams provides some new extensions, modifications and simplifications which should help to model the business processes. The processes which are modeled by this technique should be easy to construct, easy to read and also possible to verify. The preliminary research and basic introduction to this technique is mentioned in this paper.

Keywords: Business Process Modeling, BPM, UML, Activity Diagrams, Formalization

1 Introduction

The importance and role of the business processes and their modeling are still growing. The utilization of business processes modeling is not concerned only about standard software process but it can be applied also to the business and management areas. For that reason, it is important to offer the tools and methods which combine facilities and possibilities of process modeling techniques on the one hand and requirements consequent on the demands or potentialities of "non-informatics process's user or business user" on the other hand. This paper describes the technique that was developed for bringing the modern modeling methodology closer to the real business process with an emphasis on the understandability and possibility of formalization. This is the important difference in comparison with other business process modeling approaches based on UML. These goals are achieved by improving *Activity Diagrams* with extensions, modifications and some simplifications.

The main prerequisite for understanding of this approach is knowledge of the business process definition. The *Business Process* (BP) is partially ordered set of the procedures and activities, which together realize business or strategic goals. The BP also describes the relations between activities and actors, and defines their functions and responsibilities [1].

There are many tools that can be used for purposes of the business process specification[3]. The author of process model can encounter some difficulties during the modeling with these tools. The first issue is that not all of these tools can be used for correct and unambiguous definition of the business processes. This embarrassment comes from the level of formalization of each modeling method. The ability of formalization is very significant for next verification

* Department of Computer Science, FEI, VSB – Technical University of Ostrava, 17. listopadu 15, 708 33, Ostrava-Poruba, Czech Republic

and manipulation of the modeled processes. The second issue is a readability of the business process model for "standard users". The methods and their models, which are strictly based on such formalism (e.g. Petri Nets), seem to be too complicated for them. Therefore, the real power of modeling facilities is in the combination of graphical, textual and mathematical languages within one model. The graphical and textual language is suitable for the understandability whereas the formalization is suitable for the transformation and verification of processes represented by these models. The technique (called *UML-like Business Process Modeling*) mentioned in this paper is based on this approach.

2 Usage of Activity Diagrams for Business Process Modeling

The one of typical examples of graphical based modeling methods is an *Activity Diagrams* defined by UML. This technique is mainly user oriented without the real formalization behind it. It offers relatively good readability of models but it suffers from the inconvenient verification. The UML Activity Diagrams were primarily designed to model processes as a part of software process. The usage of this technique for the business process modeling purposes is also possible. But it is necessary to make some extensions, modifications or simplifications of these standard UML methods for the real effective modeling of business processes [2]. The result of these improvements should be a *UML-like Business Process Modeling* (UML-like BPM) technique.

The exactness, lucidity, effortless reading and smooth and fast construction are the main properties that are required for correct and well-designed model of business process. Compliance with these requirements particularly depends on the facilitation of process model development by the modeling application. This development framework should provide high user comfort during the process model creation, exploitation and following processing. The layer support, the filtration of presented information and automatic adaptation of diagrams are some functions related to the user-friendly approach.

3 Modeling Technique based on Activity Diagrams

The possibilities and knowledge of standard UML Activity Diagrams construction [5] is the ground for this UML-like BPM technique. The accomplishments are required because the UML-like modeling technique uses the basic mechanisms and frameworks defined by the UML. After all, there are a couple of new features related to the problems mentioned above.

First of all, the "*Swim lanes*" (that are the graphical representation of responsibility) are totally deprecated in the UML-like BPM technique. In the Activity Diagrams, the swim lanes are a kind of package for organizing responsibility for the activities. The expression of responsibility is not enough for the modeling of business processes. But also the actors' participation on a given activity must be defined. The graphical management of swim lanes and their activities is also often inconvenient due to the necessity of ordering of the activities into the swim lanes layout. This problem is solved by movement of such information (responsibility and participation) inside the activity element.

The biggest change is concerned with the activities. The activities in the UML-like Business Process Models can describe the elementary actions or they can contain complex definition of internal behavior (displacement of "sub-process element" that is deprecated too). The activity element is also the holder of the information about responsibility and participation of the roles on the activity firing and also about the input objects or some output scenarios together with a set of output objects. The more accurately definition of the activity element is mentioned below.

The analyst has to keep some rules during the process of creation of the standard Activity Diagrams. Unfortunately, the rules defined by UML are relatively lax. This "freedom" is a reason why the correct formalization and following verification of Activity Diagrams are very hard. The rules defined by mentioned technique are more subtle. The result of keeping these rules is a well-formed model that is able to formalize, transform to other languages or verify.

4 Business Process Model

The mentioned technique for specification of the business processes sees the business process as a complex model rather than a set of some diagrams. It means, that the product of UML-like BPM technique is a detailed description of business processes and their components. The output of it is a *BPM Model.* The BPM Model, as well as its components, should be defined by some formal methods:

A BPM Model is a four-tuple (D, A, O, Ac).

Let's look at single elements of BPM Model as well as their graphical representation in the process diagrams.

4.1 Elements of the Business Process Model

4.1.1 Diagrams (D).

The diagrams model the business process as a map of the activities and other nodes. Each diagram from the set D is assigned to the one activity (each activity can have maximal one diagram). It describes the control flow of the business process related to a given activity. The development of these diagrams is derivate directly from the UML Activity Diagrams approach and it is complemented by rules for well-formed modeling.

4.1.2 Actors (A).

Each business process has a set of actors which are related to it in the real life. The actor can be a human, machine or other information system that act in some role or roles. The actor is defined by its unique name and a set of these actors is a elementary part of BPM Model.

4.1.3 Objects (O).

Some things are consumed, produced or manipulated during the business process execution. The "objects" are used to describe these things, e.g., documents, products, cars or computers. The object is defined by its unique name, and the set of these entities is a part of a BPM Model again. Each object can have some attributes or operations that can provide some basic actions.

4.1.4 Activities (Ac).

The activities and their connections are the most important components of whole model. The activity is the fundamental unit of executable functionality. The aim of the activities is to define actions during the process execution. Usually, the activities have someone who is responsible for their execution and also someone who participates in them. The activities consume, produce or manipulate objects. All of these items can be subsumed under the activity element.

An Activity is defined like a five-tuple (n, ra, PA, IO, OS), where

- n (name) is a unique identification string of a given activity,
- ra (responsible actor) is one of the actors (A) that is responsible for the activity execution,
- PA (participated actors) is a finite set of actors (A) which participate in the activity,
- IO (input objects) is a finite set of objects (O) which are consumed by the activity,
- OS (output scenarios) is a finite set of scenarios.

A "Scenario" is a couple (sn, SO), where sn (scenario name) is a unique identification of scenario within a given activity and SO (scenario objects) is a finite set of objects (O) assigned with a given scenario.

Each activity can be finalized by different situations that is described by scenarios. These scenarios correspond to sets of objects which are produced by a given activity. The scenario approach is used for design simplification and for the well-formed models obtaining.

Finally, the "main activity" has to be defined in the BPM Model (it is a member of Ac). This activity corresponds with the process that is primary process of whole model. It is mapped onto a "main diagram" (it is a member of D). The diagram presented in the example (fig. 1) is the diagram that describes behavior of such activity – "Computer Selling".

There are some rules which extend definitions mentioned above.

- $ra \in A$ or $ra = \epsilon$ – activity may or may not have a responsible actor; responsible actor is also someone who participates on the activity, but it is not a member of PA set,
- $PA \subseteq A$, $|PA| \geq 0$ – activity may or may not have any participated actors,
- $IO \subseteq O$, $|IO| \geq 0$ – activity may or may not have any input objects,
- $|OS| \geq 0$, iff $|OS| = 0$ then the activity has no output scenarios as well as objects, iff $|OS| = 1$ then the output objects of a given activity are identical with output objects of this "default" scenario, in other cases the output objects is defined by scenarios.
- $activity\_diagrams : D \rightarrow A$ is a function which maps each diagram onto an activity whose internal behavior is modeled by this diagram.

4.2 Graphical Nodes of the Business Process Diagrams

The preview of elements that are used in this technique is specified above. But the definition of elements graphical representation is also necessary.

Each diagram is the representation (formal and graphical) of the relations between activities which realize control flow of a given process. The graphical nodes used to construct these diagrams are very similar to the graphical representations of UML Activity Diagrams elements. The biggest differences between this one and UML approach are in the graphical representation of initial, final and activity elements and in the deprecation of "Swim lanes" and process elements.

Figures 1 and 2 illustrate an example of *Business Process Model*. This example describes the processes of "Computers Selling" and it contains two specification of behaviors within the activity, it means that diagrams are depicted there. Some approaches and possibilities of BPM are introduced in this “demo model” (the ways of information interpretations, the well-formed approach, etc.).

All of possible forms of activity nodes are illustrated there. The most of activity nodes are displayed in the full form with all information related to these activity. Some activity nodes are presented in dynamic style where the icons (person or sheet of paper) supply list of properties. These activity properties can be displayed as floating objects or as a textual tables which complement the graphical expression of business process. This example also contains the basic algorithm structures (sequence, branching, parallelism and loop). Of course, there is able to use notes too.

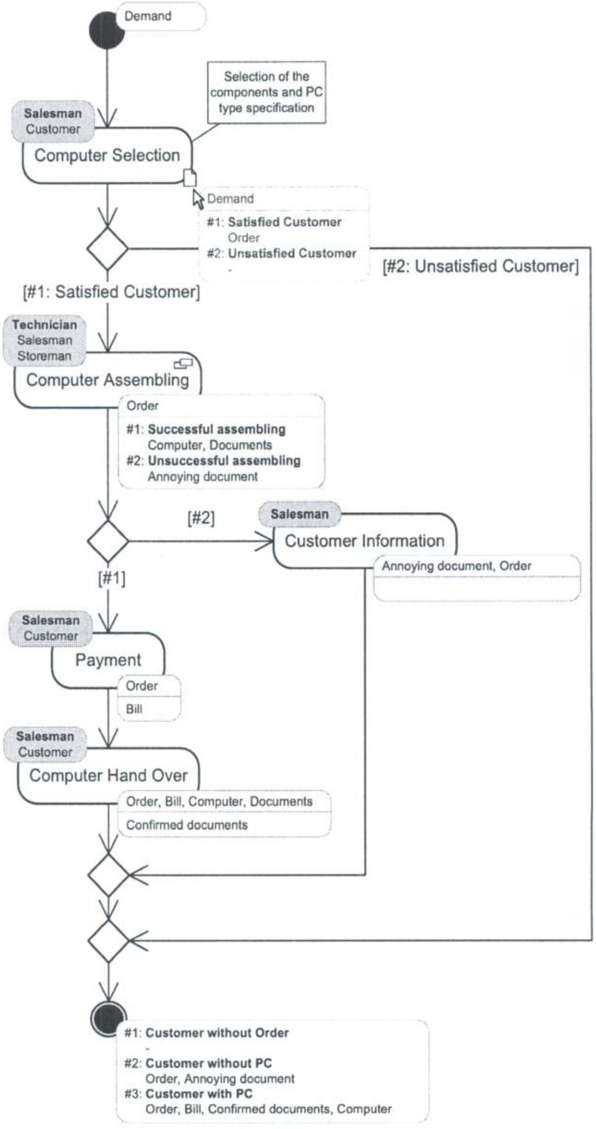
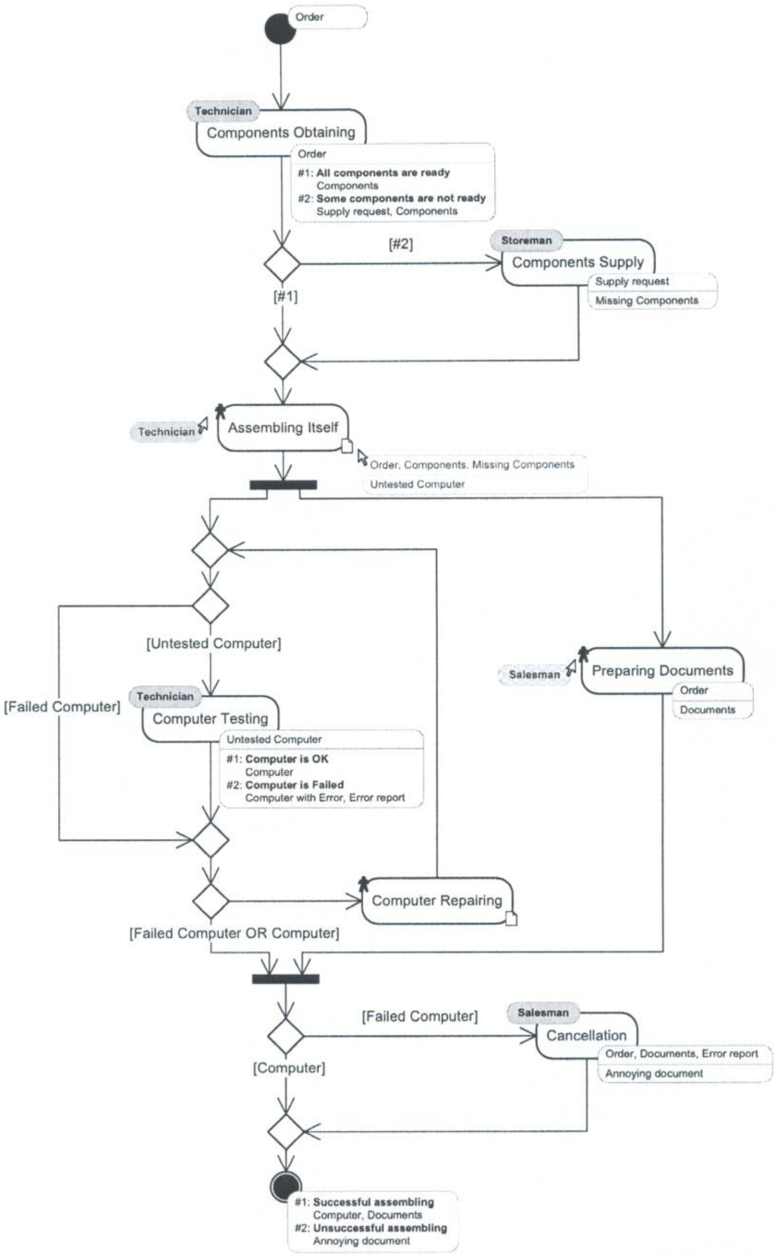


Fig. 1: The diagram of primary process that describes an example of “Computer Selling” process modeled by UML-like BPM.



Activity	Responsibility	Participation	Input Objects	Scenarios and Output Objects
Computer Repairing	Technician	-	Computer with Error, Error report	**#1: Computer is repaired** Untested Computer **#2: Computer is unable to repair** Failed Computer, Error report

Fig. 2: Diagram of internal behavior of "Computer Assembling" activity.

4.2.1 Edges.

The edges express the possible paths of control flow and connect the elements of BPM Model together. The edges are represented by oriented lines or arcs and they are used for connecting several elements of diagram. The usage of edges is without any limitations and they can connect all types of graphical nodes. The edges outgoing from the decisions nodes are the special cases that are mentioned in the next sections.

4.2.2 Activity node.

This type of nodes represents the activity element and it is completely remade in comparison with activity node used in the UML Activity Diagrams. The example of this node with description of it components is showed in Figure 3.

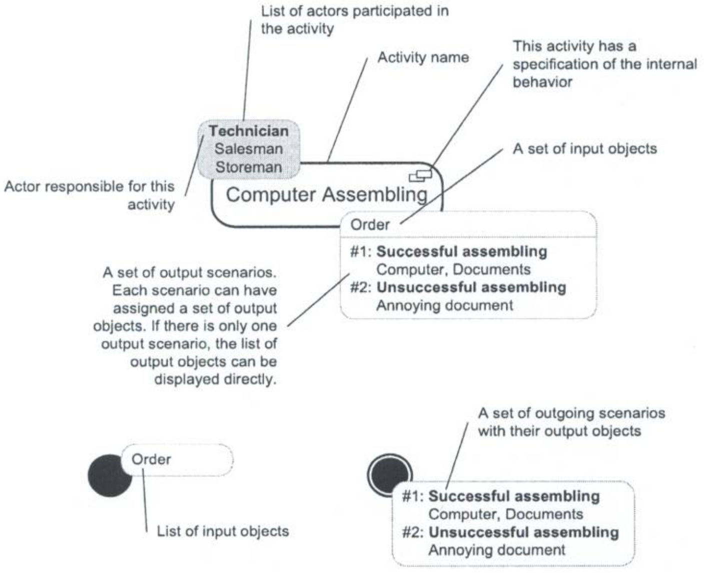

List of actors participated in the activity
Activity name
This activity has a specification of the internal behavior
A set of input objects
Technician
Salesman
Storeman
Actor responsible for this activity
Computer Assembling
Order
#1: Successful assembling
Computer, Documents
#2: Unsuccessful assembling
Annoying document
A set of output scenarios. Each scenario can have assigned a set of output objects. If there is only one output scenario, the list of output objects can be displayed directly.
A set of outgoing scenarios with their output objects
Order
List of input objects
#1: Successful assembling
Computer, Documents
#2: Unsuccessful assembling
Annoying document

Fig. 3: Graphical representation of activity element and initial and final nodes

This graphical node can be adapted in terms of user demands. It means that the information contained in the activity element can be showed by various ways based on the combination of graphical, textual and dynamical style of data presentation. This capabilities is presented in the example at the Figure 1.

The activity nodes can have only one input and one output edge due to keep "one initial and one final node" rule[4]. The edges connected to the activity node correspond to the start or final points of the activity firing as well as to the initial and final nodes of the activity itself. The implementation of scenarios is necessary for assurance of just one final node existence.

4.2.3 Initial and final node.

Initial and final nodes are graphical shapes which determine the start and final points of the process control flow. Each diagram has to have only one initial and one final node due in order to possibility of well-formed model construction. The information about input objects (in the case of the initial node) and information about output scenarios or objects (in the case of the final node) can be presented in the internal behavior diagram of a given activity.

4.2.4 Decision and merge node.

These nodes are presented by diamonds and their task is to specify a branching of control flow. The determination about next path in the process execution is provided by the decision nodes.

This branching is based on the outgoing edges conditions. These conditions can be expressed as a simple text statement or they can be specified as a logic formulas. The logic formulas can contain members of O and members of OS from an immediately preceding activity and of course the logical operators. The demands for well-formed models require the state where each decision node has a corresponding merge node at the same level. It means that no crossing arcs are in the diagram.

4.2.5 Fork and join node.

These two node types are identical with graphical nodes defined by UML and they are designated for modeling of parallelism. But the demands for well-formed models require the placement of these fork and join nodes at the same level, similarly as in the case of above-mentioned decision and merge nodes. It means that all of the possible control flows must go throw the both of these nodes.

4.3 Structure of Diagrams and Models

The well-formed diagrams should not contain any problem of critical section such as deadlocks, life locks or unreachable states. The verification of some properties (e.g. reachability, liveness, soundness, etc.) [4] is possible thanks to the formalization of this BPM Models. So, the essential conditions for the well-formed models obtaining are the existence of only one initial and one final node and the compliance of "no crossing lines" rules within each diagram.

From the structural point of view, there is no "*sub-processes*" defined by UML Activity Diagrams. Because each activity of the model can have specification of its internal behavior defined by a separate diagram, the "sub-process" term is not useful. Each activity is theoretically able to expand to other diagram. This expansion depends on the level of nicety of the business process model and its realization. If some activity does not have any specification of its internal behavior, this activity is allowed to be an "*atomic activity*" with atomic action firing. This fact is necessary for simulations of the processes and also for their transformations to the other forms, e.g., Petri Nets (atomic activity matches to the transition, activity with internal behavior matches to the complex structure of places, transitions and edges).

Another important feature of the BPM technique is the combination of textual and graphical presentation of information saved in the model thanks to the formalized background of it. The power of this technique is in simple changeover between different forms (graphical and textual) of presentation (fig. 2). Also the layering, filtration, searching and the next manipulation with models, and with information within them, are able to do with the cooperation of a proper application frameworks.

5 Future Work

The preliminary ideas and specifications are done and published at present. The clear and precise formalization of models and technique is the next step to provide a groundwork to explore extensions and usage of this method. The mapping of the UML-like BPM onto the Petri Nets could be a good way to verify the diagrams, but the precise formal transformation is necessary to be given, at first. This is one of the future plans of this research. Thanks to this, the diagrams will be able to verify by the existing algorithms and principles defined in the Petri nets theory domain.

The goal of this research is also to develop the software application and framework for creation, design and manipulation of the business processes. This application should support all above mentioned features and properties. It will be implemented with emphasis on the user-friendly environment, easy work with the models and formalized background.

6 Conclusion

In this paper, the introduction and specification of the Business Process Modeling technique based on the UML Activity Diagrams was described. The modeling approaches, elements, graphical nodes were mentioned along with a preview of future work.

The real power of this UML-like BPM technique is in the combination of theory and software application. This integration is essential prerequisite for effective, simple, correct and beneficial usage of this technique in the real life business processes modeling.

This research has been supported by the program "Information Society" of the Czech Academy of Sciences, project No. 1ET101940420 "Logic and Artificial Intelligence for multi-agent systems"

Bibliography

1. Radecky, M., Vondrak, I.: Formalization of Business Process Modeling, ISIM 2005. Hradec nad Moravici 2005. (2005)
2. Radecky, M., Vondrak, I., Stolfa, S.: Business Modeling and Its Possibility of Practice Utilization (in czech). Information systems in practice 2004. Ostrava. (2004)
3. Radecky, M., Vondrak, I.: Modeling of Processes. WOFEX 2005, Ostrava (2005)
4. van der Aalst, W.M.P.: Verification of Workflow Nets. In: P. Azema and G. Balbo, editors, Application and Theory of Petri Nets 1997, volume 1248 of Lecture Notes in Computer Science, pages 407-426. Springer-Verlag. Berlin. (1997)
5. Pender, T.: UML Bible. Wiley 2003. ISBN: 0764526049. (2003)
6. Penker, M., Eriksson, H. E.: Business Modeling With UML: Business Patterns at Work . Wiley 2000. ISBN: 0471295515 (2000)