A Meta Model Based Extension of BPMN 2.0 for Mobile Context Sensitive Business Processes and Applications

Julian Dörndorfer[1] and Christian Seel[1]

[1] Hochschule Landshut, Computer Science, Landshut, Germany
{julian.doerndorfer,christian.seel}@haw-landshut.de

Abstract. Smart devices like smartphones or tablets have become ubiquitous, which affected many daily work activities like maintaining contacts via a mobile CRM anywhere, anytime. Thus, business processes can now be executed independently of an employee's location. In addition, mobile devices have the possibility to measure physical quantities through sensors, like location or acceleration. Moreover, the connection to wireless networks made it possible to query context information like customer history. These context information can be used to adapt mobile business processes and the mobile application that support them. But in order to use this advantage, mobile sensor data has to be reflected in the business process model. As current languages for process aware information systems, such as BPMN, do not support the influence of mobile context information, we propose an extension of the BPMN that will enable the modeling of mobile context sensitive business processes.

Keywords: Business Process Modeling, Context, Mobile Business Processes, DSML, BPMN Extension

1 The Impact of Context on Mobile Business Processes

The modeling and implementation of business processes is a standard approach in Information Systems theory and practice [1–5]. In the last few years, mobile devices like smart phones, tablets, wearables, etc., affected both the work of employees and employers and thus the modeling and conduction of business processes. In a survey by Intel Research [6], 47% of employees stated that they used smart phones for their daily tasks in the workplace. 18% of the respondents use tablets for their daily job. New issues like bringing your own device [7, 8] and mobile device management [9] illustrate the impact of using mobile devices in the workplace. For field worker, who visit customers, mobile device are even more common. This change from stationary computers to mobile devices also concerns business processes. Processes, which are supported by information systems became nomadic and can now be executed anywhere and anytime. However, not only the location of the execution has changed, but the sensors of mobile devices can also deliver valuable data that affect business processes. These data can be used to directly adapt business processes to the current context that is detected by sensors. Current sensors usually measure location, acceleration, brightness and, depending on the device, other physical quantities. In addition to

13th International Conference on Wirtschaftsinformatik,
February 12-15, 2017, St. Gallen, Switzerland

Dörndorfer, J.; Seel, C. (2017): A Meta Model Based Extension of BPMN 2.0 for Mobile Context Sensitive Business Processes and Applications, in Leimeister, J.M.; Brenner, W. (Hrsg.): Proceedings der 13. Internationalen Tagung Wirtschaftsinformatik (WI 2017), St. Gallen, S. 301-315

connecting to the mobile or wireless network, they can query data from different information systems or other machines [10]. Therefore, mobile devices are capable of measuring context and evaluating it via algorithms. Thereby, DEY defines 'context' as any information that can be used to characterize the situation of an entity [11]. An entity can be a person, place or object which is considered relevant to the interaction between a person and an application. Context can be used to pre-select services to reduce the information overload for a user [12].

The various possibilities of mobile devices can support the user by making decisions in a business process or when carrying out a task. FALK and LEIST [13] were able to show that mobile applications have positive impact on business processes, like cost and time reduction, as well as quality and flexibility improvements. Furthermore, applications which measure and interpret the context can automatically execute parts of the business process without the involvement of any other party. Additionally, they can adapt the process, for example skip or block [14] an activity, and influence the execution. A good example are the employment protection inspectors. From time to time they have to control the working condition of the employees, the correct montage or installation of machines and their proper functionality. In some countries like Germany these regular inspections are prescribed by law like the 'Betriebssicherheitsverordnung' [15]. The inspectors have to check and to document different things depending on their location, time or presence of employees. If the inspectors are in an assembly hall, they could check the machines or, if the machines are running, they could control the work clothes. If the assembly hall has security areas, they could control the security mechanisms and precautions for the employees. All these tasks are mainly depending on context information, which can easily be accessed by smart devices. They also support the task executions by showing what to control and by documenting completed checkups. All these process steps are immediately interrupted if the context switches to emergency because of an accident.

This mobile context sensitive business process gives a good impression of how context data can be used to support and ease the work of employees and increase the efficiency of business processes. However, to plan mobile context sensitive business processes a modeling language has to support context. The standard languages in the area of information systems, like Business Process Model and Notation (BPMN), Event-driven Process Chain (EPC) or the Unified Modeling Language (UML), do not support the modeling of context [16–18]. However, some approaches were made to extend the standard modeling languages to improve the configuration and flexibility of them (cf. section 2), these approaches where neither designed for mobile business processes nor mobile sensor data.

As BPMN 2.0 [19] became the de-facto standard [20] and allows to automate business processes based on execution engines, it seems reasonable to extend BPMN with adaption to mobile context. Hence, the following research question (RQ) arise:

RQ.1: *To what extend could the standard elements of the BPMN 2.0 specification be used to model contextual influences?*

RQ.2: *How could extension of the BPMN be structured to ensure an appropriate modeling of mobile context sensitive business processes?*

The remainder of this paper is structured as follows: Section 2 introduces a brief overview about existing approaches and related work in this field. In the beginning of section 3 the existing BPMN elements are analyzed for the capability to express context (**RQ.1**). Thereafter, the requirements for the extension are discussed and collected. Section 4.2 presents a meta-model based BPMN extension (**RQ.2**). Afterwards an example model is shown in order to prove the applicability of the BPMN extension. The paper ends with a conclusion and outlook to further research in context sensitive mobile applications.

2 Related Work and Research Method

The term context was defined among others [21–23] by DEY [24]. DEY'S definition is well known and accepted in the scientific community. He also declares an application as context-aware "if it uses context to provide relevant information and/or services to user, where relevancy depends on the user's task" [24].

Several approaches have been conducted to integrate context into business processes. ROSEMANN et al. [25] claim that modeling languages have to be more flexible to model context. Further they state that an increased attention on flexibility took place in the research area, which leads to a decreasing time-to-market for products [26]. Therefore, the result is a demand for higher process flexibility [27]. In particular, ROSEMANN et al. show the limitation of the actual EPC language and the lack of supporting context modeling. They also present a framework that helps to understand the different types of context and their impact on business processes. In addition, they categorize the different influence factors of a business process. However, they do not present an appropriate way to integrate the identified context in a business process. C-EPC is an extension of EPC to make it more configurable for decisions at runtime [28]. In [29] an approach to identify all variants of a business process depending on its context is presented. LA VARA et al. aim to reveal all possibilities of a business process and integrate them into one model. The outcome is a large and complex model. A way to identify and apply context on business processes is introduced in [30]. However, it is more of a theoretical framework to identify context. But the authors do not show how a context sensitive business process could be designed.

Furthermore two extensions for UML are published by AL-ALSHUHAI and SIEWE. In [31] they are extending the class diagram with additional annotations. The second paper [32] expands the activity diagram to mark context sensitive areas or sequences.

HEINRICH and SCHÖN [33] mention that business processes must consider „non-static" context events which change the process conduction, like an upcoming thunderstorm in an outdoor process. They further present an algorithm which supports automated process planning for context-aware processes, but no modelling representation in BPMN 2.0. CONFORTI et al. [34] presented an approach to cope with process risks based on sensor evaluation. Furthermore, they presented a way to model these risks and when they occur. However, they see sensors only as a source for risk evaluation, but context and its evaluation via sensors is more than only risk analyzation for business processes.

All of the presented papers do not consider how to model mobile context sensitive business processes. Therefore, this paper aims to present a new context extension for BPMN and creates a new artifact based on the design research approach by HEVNER et al. in [35]. Therefore, the meta-model of the extension will be created with Meta Object Facility (MOF) 2.0 [36]. New notation will be developed and a guideline to assist the user presented.

3 Requirements on Context Modeling of Mobile Business Processes

3.1 Analysis of the Existing BPMN Elements

To answer the **RQ.1** the BPMN 2.0 core elements will be examined in this section. They offer a few possibilities to model contextual mobile business processes. These possibilities consist of all the existing gateways (exclusive, inclusive, complex etc.) or intermediate events. Modeling context in mobile business processes using only gateways leads to a large and confusing model (cf. figure 1). Intermediate boundary events, which could be attached on activities, would be another reasonable option to model context influences. However, no event exists that really matches the description of context. For example the intermediate event *Message* “can be used to either send a Message or receive a Message” [18]. It can be used to symbolize that a context change “sends” an information, which leads to a reaction. At a first glance this looks like a promising method to model context, but a closer look reveals the problem. The event *Message* is, like other alternative events, very generic and could have several meanings. In addition, there is no possibility to describe a proper context expression. The result of using this event for depicting context influences would be a large documentation, which is counterproductive to the aims of a process visualization. Moreover, existing BPMN notation cannot sufficiently cope with non-static context [33], like changing parameters at runtime. A mix of intermediate events and gateways seems to be even worse. Confusing models with a documentation is not the idea of business process modeling. The result of the analysis is, that the existing elements of BPMN cannot be used to model context.

However, the specification of the BPMN offers a possibility to extend existing elements or add new elements to it [18]. To develop an extension of the BPMN, the requirements have to be imposed and discussed.

3.2 Requirements for the Extension

As described before, no explicit symbol or label exists which marks that an activity is influenced by context. Furthermore, not just one activity but a whole process sequence could depend on context data. In the example from section 1 an employment protection inspector has been introduced. She or he controls the working and security conditions at the workplace and inspects machines.

Figure 1 depicts a simplistic inspection process. Depending on her or his location (office building or different assembly halls) and present workers, she or he controls the first-aid kit or the security mechanisms. However, it is difficult to see that an activity or sequence depends on context information. Many decisions have to be made and modeled by gateways, which leads to large and complex models. Thus, the first requirement for the extension is an element which marks an activity or a process sequence depending on the context.

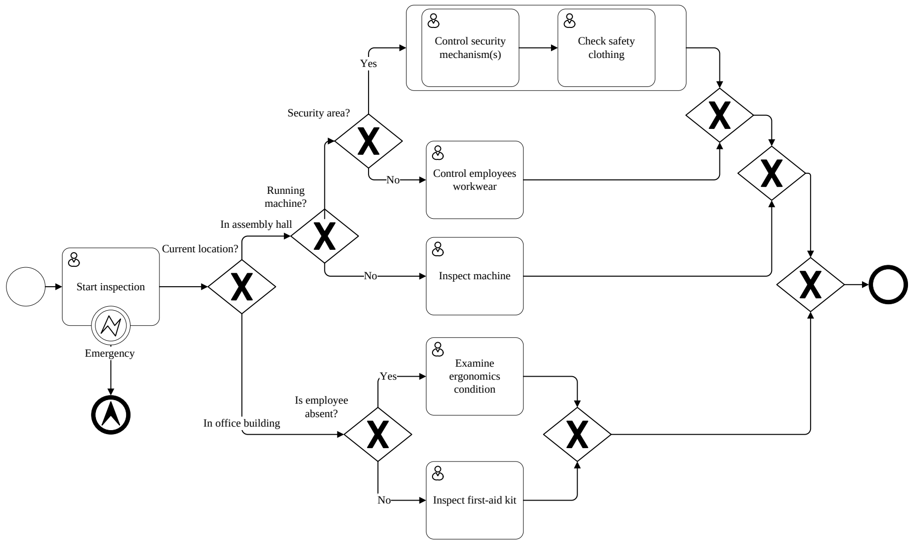


Figure 1. “Employment protection process” with BPMN 2.0 core elements

BPMN events are divided into interrupting and non-interrupting. Also a changing context can be an interrupting or non-interrupting event for an activity or sequence. For instance, if an emergency happens in the assembly hall, the inspector would take suitable measures and she or he would interrupt his or her current activity. At the moment, it is hard to model such an occurrence, because every activity in the process would have an attached exception event, like the one depicted in figure 2.

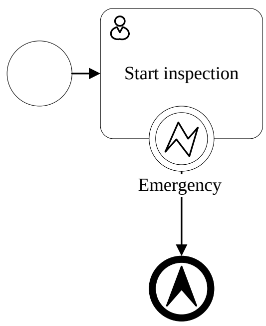


Figure 2 – Example for exception modeling in traditional BPMN 2.0

This would inflate the model unnecessarily. However, not every context change leads to an interruption of the current activity or sequence. Let us assume that a shift change happens during the first-aid kit check. Nevertheless, the context event “shift change” does not interrupt the activity “inspect the first-aid kit”. Therefore it is required that context events can be interrupting or non-interrupting. Another requirement is to express context dependencies in a short but clear way. A decision can depend on more than one context data. A textual description could lead to long sentences or even to an extra documentation file. To avoid this, a brief description method should be developed and applied.

The time to evaluate the context is important. Depending on the concrete business process, two possibilities exist. First, the context measurement is necessary at one explicit point in time. The evaluation of the context “location” only needs to be queried at the beginning of the inspection, because the location remains unchanged. In other processes the context information will be evaluated continuously. The context information “emergency” has to be queried in a tight frequency (e.g. 2 seconds) to show supportive information on the smart devices. It would be impractical to query this information only once at the beginning. Therefore, it must be possible to express the time of evaluation in the mobile business model to cope with non-static context.

In section 1 the advantages of using a standard modeling language have been discussed. To utilize these advantages the extension has to be compatible to the BPMN 2.0 standard. Thus, this is an additional requirement for the extension.

Context is measured through sensors. We define sensors not only as devices that detect and respond to a physical in- or output, like a hygrometer or a temperature sensor. It could also be a database or an application from which information could be requested or even any machine that is accessible via a network connection. In addition, a context information could be based upon different sensors. For example the context information “weather” is based on the following sensors: “temperature”, “humidity”, “condensation”, etc. Furthermore, the context information “weather” could be a “sensor” for the context “traffic”, in order to warn of snow in a traffic information system. So, context and its sensors could have interdependencies. This shows that context information is hard to model due to its complexity. Therefore, we decided to split the sensor modeling from the BPMN model and introduce an extra model. A link will be established to connect these two models. All the discussed requirements of this section are summarized in Table 1.

Table 1 - Requirements for the extension

Requirement	Description	Model
Context marker	A way to mark an activity or a sequence as context depending.	BPMN extension
Context event is interrupting / non-interrupting	A context event can be interrupting or non-interrupting.	BPMN extension

Requirement	Description	Model
Context expression language	Express a context condition in a brief way.	BPMN extension
Evaluation point	The point in time at which the context information should be evaluated.	BPMN extension
Compatibility to BPMN 2.0	The extension has to be compatible to BPMN 2.0.	BPMN extension
Sensor modeling	A possibility to model the dependencies of a context information to sensors or other context information.	Dedicated DSML for sensor modeling

4 Meta Model Based Extension

4.1 The BPMN Extensibility

A common way to extend modeling languages is a meta model extension [37]. A meta model is besides a notation and a guideline ([38] p. 36) [39] the first step to develop a complete modeling technique. A meta model is a model (M2) which describes a language (L1) to create a model (M1) of an object [40]. BPMN describes all of its elements in a class diagram, which is the meta-language for it. In addition, the BPMN specification itself describes a mechanism to extend the model to the needs of domain ([18] p. 57). The mechanism ensures the core validity and enables a way to integrate domain specific concepts. It consist mainly of four classes: Extension, ExtensionDefinition, ExtensionAttributeDefinition, ExtensionAttributeValue. These classes allow to expand existing elements through additional attributes or to enhance the BPMN meta model. To meet the compatibility requirement to BPMN 2.0 we use this well described mechanism to extend the BPMN model.

4.2 Meta Model of the Context Extension

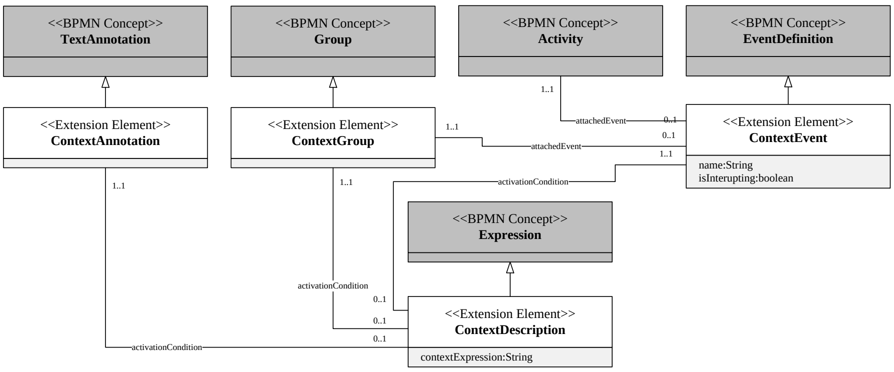


Figure 3. Meta model of Context4BPMN

Based on the in section 3.2 described requirements a meta model for Context4BPMN has been developed. The result is depicted in figure 3. Object Management Group's (OMG) standard BPMN classes are colored in grey. To mark context sensitive areas in a BPMN the Group class is extended by the *ContextGroup* class. The *ContextEvent* class creates the new context event which can be interrupting and non-interrupting as well as attached to an activity or a *ContextGroup*. Whether it is interrupting or not, is represented by the boolean variable. The *ContextDescription* class inherits from the *Expression* class to allow expressing the activation conditions of an activity or sequence. A *ContextDescription* includes the *contextExpression* to define the activation conditions, which are stored in a string. The grammar of the expression is explained in section 4.3. The *ContextDescription* could be used by the *ContextAnnotation*, *ContextMarker* and *ContextEvent*.

4.3 Syntax of the Context Extension and Example

In figure 4 the new element's graphical notations of the Context4BPMN extension are presented. The first two elements are the *Intermediate Context Events*, which add two new event types to the *Event* elements. The first element with the two solid circles and the eye is the *Interrupting Context Event*. Thereafter comes the *Non-Interrupting Context Event*. Both can be attached on *Activities* or *ContextGroups* concluded from the meta model. The *Context Annotation* is an enhancement of the *Text Annotation*. A little icon in the upper inner corner marks this new element. It can be attached to nearly every element like the "original" *Text Annotation*. The last element is the *Context Group*, which marks a activity or a sequence as context depending.

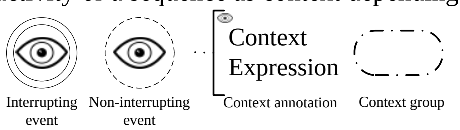


Figure 4. The new notations

```
<ContextExpression>::= [<EvaluationPoint>";"]<ContextTerm>
<ContextTerm>::= <Variable> <Comparison> <Value> |
  <ContextTerm> <LogicOperator> <ContextTerm>
<Comparison>::= "=" | "!=" | "<=" | ">=" | "<" | ">"
<LogicOperator>::= "AND" | "OR" | "XOR"
<EvaluationPoint>::= "Evaluation Point = continuous @"
  <Interval>| "Evaluation Point = " <Time>
<Time>::= -->"TimeDefintion according to RFC 5322"
<Interval>::= -->"IntegerNumber" "msec" | "sec" |"min" |
  "hours" | "days"
<Variable>::= -->"StringIdentifier in UTF-8"
<Value>::= -->"StringIdentifier in UTF-8"
```

Figure 5. EBNF Grammar for context expressions

In order to express context in a structured way that can be evaluated automatically a context expression language is developed. The context free grammar for this language has been defined in the extended Backus–Naur Form (EBNF) [41] in figure 5. To shorten some basic definitions, like integer numbers or date, we link with "-->" to standards, like the formal definition in RFC 5322 for date formats. A context expression contains a context term, which basically consists of "variable" "comparator" "value". The "variable" is the name of the context, like "location" or "status". It has to be unique in a business process and is also the link to the sensor model declaring how to measure it. Examples of the language are presented in figure 8.

4.4 Guideline for the Extension

To get a complete modeling technique a guideline for the modeler to apply the language extension is necessary ([38] p. 36) [39]. Three initial situations of mobile context sensitive modeling exists. First, a new process has to be modeled from scratch. Second, an existing process has to be remodeled as context sensitive. The last situation is when different variations of a business process depending on context variables exist and have to be merged into one business model. The first two situations differ only in the beginning, when a new business process has to define its activities, whereas an existing process has to be decomposed. Hereafter, the next steps are identical. Figure 6 and figure 7 are depicting the guideline for the different situations, which are modeled in BPMN. The guideline for the first and the second situation are an expanded version from the "procedure for context identification" by ROSEMANN et al. [25]. If a process exists, a decomposing would be necessary to get a list of activities, whereas for a new process a procedure has to be defined. At the end of both tasks a list of activities would be the outcome. The next step would be to determine context influences. The aim of this step is to identify the relevant context, which will be used in the next step to detect the context depending activities. Hereafter, a mobile context sensitive business process model could be developed with the identified context depending activities.

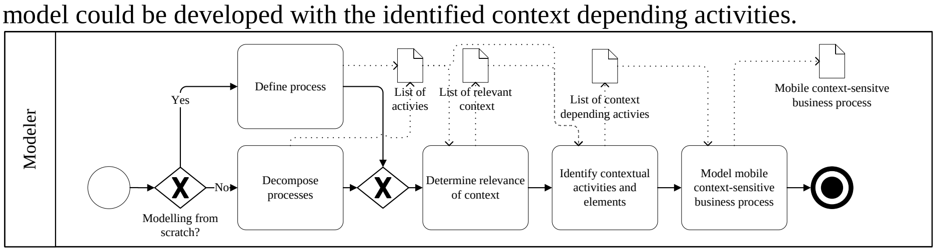


Figure 6. Guideline for initial situation one and two

If different variations of the same business process exist, the first task has to determine all variations. The output, which is a list of a variations, will be used to identify the relevant context, causing the variations. The third step is to merge all alternatives into one business model. For this task model matching algorithms can be used like the one proposed in [42]. Identical activities or sequences are context independent, whereas deviations in the models are most likely context depending. This holistic model will be used to identify the context depending activities or sequences.

These activities together with the relevant context variables will be used to model the context sensitive business process.

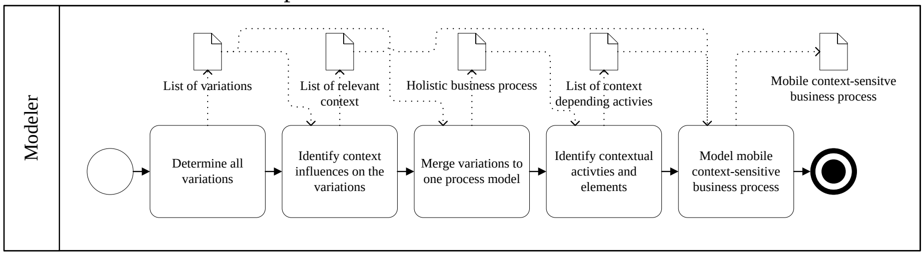


Figure 7. Guideline for the third initial situation

5 Discussion and Further Research

5.1 Requirement Examination

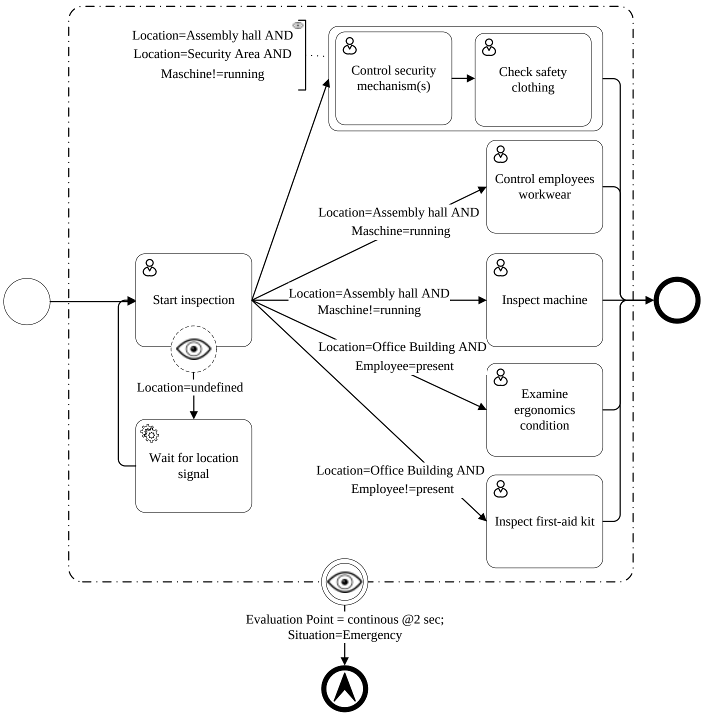


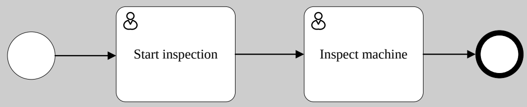
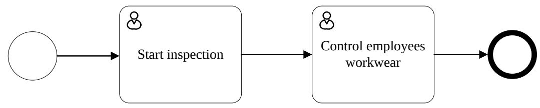
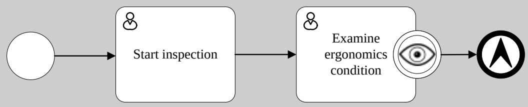
Figure 8. Inspection example depicted with the extension Context4BPMN

As stated in the beginning of this paper, it was intended to develop a possibility to model mobile context sensitive business processes. Therefore, we aimed to create an extension for BPMN to save effort and time in developing and establishing a new domain language profiting from well-known mechanisms and tool support. A brief evaluation of the created artifact as recommended by HEVNER et al. [35] will be presented. Figure 8 again depicts the example “employment protection” from section 3.2, but utilizing the new elements. The whole sequence is context depending, therefore the context group marks all depicted activities. Attached to it is an interrupting context event, which will be triggered if the situation changes from normal to emergency (non-static context). The situation could be changed at once, therefore the information is requested every 2 seconds. It would be displayed in the supporting mobile application. Further steps in the process would be defined in the catching escalation event. The “Start inspection” activity now has the non-interrupting context event attached to define the task if no location could be determined by the application.

Afterwards, one activity depending on the evaluated context will be selected. This can be seen as selecting an activity out of a pool of activities at runtime. The selection will be done by evaluating the context expression. These expressions are shown on the sequence flows and in the context annotation, attached to the brief sub-process.

Table 2 depicts the applied process steps for a certain context situation. For instance, the first row shows the process when the inspector is in the assembly hall, the machine is not running and no emergency interferes the inspection.

Table 2. Different versions of the process at runtime

Variation	Context	Applied version of the process
1	Assembly hall && machine not running && no emergency	
2	Assembly hall && machine running && no emergency	
3	Office building && employee present && emergency	

In section 3.2 a table with the requirements for the extension was introduced. To reconcile the requirements with the developed extension features we conducted a comparison.

Table 3. Comparison between the requirements and the features of Context4BPMN extension

Requirement	Feature	Match
Context marker	Context Group	✓
Context event is interrupting / non-interrupting	Intermediate Context Event	✓

Requirement	Feature	Match
Context expression language	Context expressions could be placed on flows and in Context Annotations	✓
Evaluation point	The Evaluation point can be stated in the context expression language	✓
Compatibility to BPMN 2.0	The extension uses the BPMN enhancement mechanism to secure the compatibility to the 2.0 version	✓
Sensor modeling	Due to lack of space for this paper the sensor model could not be presented. Only the link to it is established. Therefore, this feature is not completely described	-

As can be seen from table 3 all direct requirements to the BPMN extension are fulfilled and depicted in figure 8. A context marker is established as a context group element in the extension. Even though two intermediate context events were added, one interrupting and one non-interrupting, context expressions can now be stated on flow elements or on context annotations. It is also possible to express an evaluation time in the context expression language. A link to the sensor model is as well established. Hence, it can be claimed that all stated requirements are fulfilled.

5.2 Outlook

The main contribution of this article is to provide an extension to model mobile context sensitive business processes. It provides the possibility to model engineers to plan such processes in a precise, detailed and comprehensive way. It also enables to model the influence of context, like intermediate events on groups, which has not been available before.

There are some tasks for further research in this area: First of all, a way to model the relationships between context and sensors, furthermore, to model the interdependencies between two pieces of context information. The reason to prefer the creation of a new language instead of integrating it into a standard language like BPMN is that at firstly the model would become unnecessarily big and complex. Secondly, the inter-dependencies of context information and measuring by sensors is not a necessary part of a business process language.

Modeling a business process is just the first step in the business process lifecycle (BPL) [43]. The implementation, execution and controlling are the remaining steps in the lifecycle. Since mobile context sensitive business processes obviously need to measure context and are supported by an application on smart devices, an automated or semi-automated way to generate code from the business process would be helpful to increase the fluency between modeling and implementation phase. The logical context expressions can be used to generate decisions in the application program. Furthermore, the sensor model can be utilized to pre-generate classes and interfaces. In an additional step the use of the gathered context data from the execution of a business process will be investigated. They could be used to identify problems in the execution and therefore interesting for the controlling phase.

References

1. Scheer, A.-W.: ARIS - business process modeling. Springer, Berlin u.a. (2000)
2. Bichler, M., Frank, U., Avison, D., Malaurent, J., Fettke, P., Hovorka, D., Krämer, J., Schnurr, D., Müller, B., Suhl, L., et al.: Erratum to. Theories in Business and Information Systems Engineering. Bus Inf Syst Eng (2016)
3. Vom Brocke, J., Rosemann, M. (eds.): Handbook on Business Process Management 2. Strategic Alignment, Governance, People and Culture. Springer-Verlag Berlin Heidelberg, Berlin, Heidelberg (2010)
4. Hammer, M., Champy, J.: Reengineering the corporation. A manifesto for business revolution. Harper Business, New York, NY (1993)
5. Becker, J., Kugeler, M., Rosemann, M. (eds.): Process management. A guide for the design of business processes. Springer, Berlin (2011)
6. Intel IT Center: Mobile Computing Trends: Insight into Today's Workforce (2013)
7. Morabito, V. (ed.): Trends and challenges in digital business innovation. Springer, Cham (2014)
8. Kerr, D., Koch, C.: A Creative and Useful Tension? Large Companies Using "Bring Your Own Device". In: Bergvall-Kåreborn, B., Nielsen, P.A. (eds.) Creating Value for All Through IT. IFIP WG 8.6 International Conference on Transfer and Diffusion of IT, TDIT 2014, Aalborg, Denmark, June 2-4, 2014. Proceedings, 429, pp. 166–178. Springer Berlin Heidelberg, Berlin, Heidelberg, s.l. (2014)
9. Rhee, K., Eun, S.-K., Joo, M.-R., Jeong, J., Won, D.: High-Level Design for a Secure Mobile Device Management System. In: Marinos, L., Askoxylakis, I. (eds.) Human aspects of information security, privacy, and trust. First international conference, HAS 2013, held as part of HCI International 2013, Las Vegas, NV, USA, July 21 - 26, 2013 ; proceedings, 8030, pp. 348–356. Springer, Berlin (2013)
10. Fortino, G., Trunfio, P.: Internet of things based on smart objects. Technology, middleware and applications (2014)
11. Dey, A.K.: Understanding and Using Context. Personal and Ubiquitous Computing 5, 4–7 (2001)
12. Heinrich, B., Lewerenz, L.: A Novel Concept for the Usage of Mobile Information Services. In: Linnhoff-Popien, C., Zaddach, M., Grahl, A. (eds.) Marktplätze im Umbruch. Digitale Strategien für Services im mobilen Internet, pp. 319–329. Springer Vieweg, Berlin (2015)
13. Falk, T., Leist, S.: Effects of mobile solutions for improving business processes. ECIS 2014 Proceedings (2014)
14. Gottschalk, F., van der Aalst, W.M.P., Jansen-Vullers, M.H.: Configurable Process Models — A Foundational Approach. In: Becker, J., Delfmann, P. (eds.) Reference Modeling. Efficient Information Systems Design Through Reuse of Information Models, pp. 59–77. Physica-Verlag, Heidelberg (2007)
15. Verordnung über Sicherheit und Gesundheitsschutz bei der Verwendung von Arbeitsmitteln (Betriebssicherheitsverordnung). BetrSichV (2016)

16. Object Management Group (OMG): Unified Modeling Language (2015)
17. Keller, G., Nüttgens, M., Scheer, A.-W.: Semantische Prozeßmodellierung Semantische Prozeßmodellierung auf der Grundlage „Ereignisgesteuerter Prozeßketten (EPK)“. Saarbrücken (1992)
18. Object Management Group (OMG): Business Process Model and Notation (BPMN), Version 2.0 (2011)
19. International organization for standardization (iso): Information technology. Object Management Group Business Process Model and Notation (2013)
20. Braun, R., Schlieter, H., Burwitz, M., Esswein, W.: Extending a Business Process Modeling Language for Domain-Specific Adaptation in Healthcare. In: Wirtschaftsinformatik Proceedings 2015, pp. 468–481 (2015)
21. Weiser, M.: The Computer for the 21st Century. Sci Am 265, 94–104 (1991)
22. Schmidt, A., Beigl, M., Gellersen, H.-W.: There is more to context than location. Computers & Graphics 23, 893–901 (1999)
23. Schilit, B., Adams, N., Want, R.: Context-Aware Computing Applications. In: First Workshop on Mobile Computing Systems and Applications (WMCSA), pp. 85–90 (1994)
24. Dey, A.K.: Providing Architectural Support for Building Context-aware Applications. Georgia Institute of Technology, Atlanta, GA, USA (2000)
25. Rosemann, M., Recker, J.C., Flender, C.: Contextualisation of business processes. International Journal of Business Process Integration and Management 3 (1), 47–60 (2008)
26. Rosemann, M., Recker, J., Flender, C.: Designing context-aware Business Processes. In: Siau, K., Chiang, R., Hardgrave, B.C. (eds.) Systems analysis and design. People, processes and projects, pp. 51–73. M.E. Sharpe, Armonk, NY u.a (2011)
27. Soffer, P.: On the Notion of Flexibility in Business Processes. In: Proceedings of the CAiSE’05 Workshops, pp. 35–42 (2005)
28. Rosemann, M., van der Aalst, W.: A configurable reference modelling language. Information Systems 32, 1–23 (2007)
29. La Vara, J.L. de, Ali, R., Dalpiaz, F., Sánchez, J., Giorgini, P.: Business Processes Contextualisation via Context Analysis. In: Parsons, J., Saeki, M., Shoval, P., Woo, C., Wand, Y. (eds.) Conceptual modeling - ER 2010. 29th International Conference on Conceptual Modeling, Vancouver, BC, Canada, November 1 - 4, 2010 ; proceedings, 6412, pp. 471–476. Springer, Berlin (2010)
30. Saidani, O., Nurcan, S.: Towards Context Aware Business Process Modelling. In: Workshop on Business Process Modelling, Development, and Support, p. 1. Norway (2007)
31. Al-alshuhai, A., Siewe, F.: An Extension of Class Diagram to Model the Structure of Context-Aware Systems. In: The Sixth International Joint Conference on Advances in Engineering and Technology (AET) (2015)
32. Al-alshuhai, A., Siewe, F.: An Extension of UML Activity Diagram to Model the Behaviour of Context-Aware Systems. In: Computer and Information Technology; Ubiquitous Computing and Communications; Dependable,

Autonomic and Secure Computing; Pervasive Intelligence and Computing (CIT/IUCC/DASC/PICOM), pp. 431–437 (2015)
33. Heinrich, B., Schön, D.: Automated Planning of Context-aware Process Models. University of Münster, Münster, Germany (2015)
34. Conforti, R., La Rosa, M., Fortino, G., ter Hofstede, A.H., Recker, J., Adams, M.: Real-time risk monitoring in business processes. A sensor-based approach. Journal of Systems and Software 86, 2939–2965 (2013)
35. Hevner, A.R., March, S.T., Park, J., Ram, S.: Design Science in Information Systems Research. MIS Q 28, 75–105 (2004)
36. Object Management Group (OMG): Meta Object Facility (MOF) Core Specification (2006)
37. Atkinson, C., Gerbig, R., Fritzsche, M.: Modeling Language Extension in the Enterprise Systems Domain. In: 2013 17th IEEE International Enterprise Distributed Object Computing Conference (EDOC 2013), pp. 49–58
38. Seel, C.: Reverse Method Engineering. Methode und Softwareunterstützung zur Konstruktion und Adaption semiformaler Informationsmodellierungstechniken. Logos-Verl., Berlin (2010)
39. Karlsson, F., Ågerfalk, P.J.: Method configuration. Adapting to situational characteristics while creating reusable assets. Information and Software Technology 46, 619–633 (2004)
40. Mertens, P., Back, A. (eds.): Lexikon der Wirtschaftsinformatik. Springer, Berlin (2001)
41. Backus, J.W., Wegstein, J.H., van Wijngaarden, A., Woodger, M., Bauer, F.L., Green, J., Katz, C., McCarthy, J., Perlis, A.J., Rutishauser, H., et al.: Report on the algorithmic language ALGOL 60. Commun. ACM 3, 299–314 (1960)
42. Niesen, T., Dadashnia, S., Fettke, P., Loos, P.: A Vector Space Approach to Process Model Matching using Insights from Natural Language Processing. In: Nissen, V., Stelzer, D., Straßburger, S., Fischer, D. (eds.) Multikonferenz Wirtschaftsinformatik (MKWI) 2016. Technische Universität Ilmenau, 09. - 11. März 2016, vol. Ivol. , pp. 93–104. Universitätsverlag Ilmenau, Ilmenau (2016)
43. Weske, M.: Business process management. Concepts, languages, architectures. Springer, Berlin u.a. (2007)