



uBPMN: A BPMN extension for modeling ubiquitous business processes



Alaaeddine Yousfi^{a,b,*}, Christine Bauer^c, Rajaa Saidi^{a,d}, Anind K. Dey^b

^aLRIT, Research Unit Associated to the CNRST (URAC 29), FSR, Mohammed V University, Rabat, Morocco

^bHuman-Computer Interaction Institute, Carnegie Mellon University, Pittsburgh, PA, USA

^cDepartment of Information Systems and Information Management, University of Cologne, Cologne, Germany

^dINSEA, Rabat, Morocco

ARTICLE INFO

Article history:

Received 25 July 2015

Revised 10 February 2016

Accepted 11 February 2016

Available online 23 February 2016

Keywords:

Business process modeling
Ubiquitous business process
Ubiquitous computing
uBPMN

ABSTRACT

Context: Business Process Model and Notation (BPMN) is the de facto standard for business process modeling. It was developed by the Object Management Group with support of the major organizations in the fields of software engineering and information systems. Despite its wide use, when it comes to representing ubiquitous business processes, this business process modeling language is lacking.

Objective: To address BPMN's deficiency in representing ubiquitous business processes, we extend it and present uBPMN (or ubiquitous BPMN).

Method: First, we analyze the modeling requirements for representing ubiquitous business processes. Based on the requirements, we conservatively extend the Meta-Object Facility meta-model and the XML Schema Definition of BPMN as well as extend the notation. The extension, that we call uBPMN follows the same outline as set by the Object Management Group for BPMN.

Results: The proposed uBPMN not only allows for modeling ubiquitous business processes but also lays the groundwork for potentially deploying a variety of ubiquitous computing technologies. We illustrate all of uBPMN's capabilities and benefits with real-life examples.

Conclusion: uBPMN extends BPMN v2.0 with new capabilities to deal with ubiquitous computing technologies.

© 2016 Elsevier B.V. All rights reserved.

1. Introduction

Nearly a decade after its official introduction in May 2004, the Business Process Model and Notation (BPMN) gained the upper hand in Business Process Modeling [8], both in academia and business. As of June 2015, it was referenced in more than 24,000 scientific publications and 300 patents as listed in Google Scholar/Patents. When compared to other business process modeling languages, numerous studies (e.g., [41,48]) emphasize that BPMN may be considered as the de facto standard for business process modeling. Further still, it is supported by big names in the fields of information systems and software engineering (see Section 6.3 in [42]). Last but not least, its groundbreaking features

motivated the appearance of many engines supporting it such as Activiti [47], jBPM [57] and Oracle BPM [22].

Since its first release as v1.0 in May 2004, BPMN underwent three updates. Each update was introduced to allow BPMN to represent new process characteristics that were not covered by the version that preceded. The changes are thoroughly described by the Object Management Group (OMG) in each new release (e.g., changes from v1.2 to v2.0 are available in [42, p. 479]). Now, since its latest release as BPMN v2.0 in January 2011, Business Process Management has evolved a lot [15,24]. Particularly, many new process characteristics emerged that BPMN v2.0 cannot represent. For instance:

- **Example 1:** *The highway toll can be paid on the fly using the RFID (Radio Frequency Identification) tag on the car windshield without stopping at any toll plaza.* The problem here is that automatic identification and data capture of the RFID tag and content cannot be appropriately represented by BPMN v2.0.
- **Example 2:** *The taxi is assigned to the customer based on her/his current location.* In this business rule, a mechanism to collect

* Corresponding author at: Human-Computer Interaction Institute, Carnegie Mellon University, Pittsburgh, PA, USA. Tel.: +1 412 268 9378.

E-mail addresses: ae يوسف@cmu.edu (A. Yousfi), bauer@wim.uni-koeln.de (C. Bauer), r.saidi@insea.ac.ma (R. Saidi), anind@cs.cmu.edu (A.K. Dey).

and quantify the current location of the user and assign the most appropriate taxi to her/him cannot be represented by BPMN v2.0.

- **Example 3:** After capturing a sample of the music playing, the song is identified and added to the customer's music order. Here, the problem is that BPMN v2.0 cannot accurately describe audio collection and sampling.

BPMN, in its latest release, offers five types of core modeling elements; Flow Objects, Data Objects, Connecting Objects, Swimlanes and Artifacts [42]. When tackling the aforementioned examples using the notation, many challenges arise. As such, what Flow Objects, Data Objects, Connecting Objects, Swimlanes and Artifacts can be used to accurately cope with the three business rules from the examples? One may suggest using the existing core modeling elements of BPMN v2.0. For example, one idea would be to represent these new capabilities with Artifacts, such as Text Annotations (also core elements of BPMN v2.0). While this may appear to be a viable solution from a modeling and design point of view, severe problems arise when it comes to Verification and Validation (V&V) [64] of the process models, as Text Annotations cannot be validated. Data Objects were introduced to deal with static data (e.g., file, database). A Text Annotation to indicate that Data Objects can handle dynamic data (e.g., sensor data) would be meaningless to a V&V algorithm. The same argument applies for the Flow Objects. Still, there will be additional difficulties, when the goal is to reach the transformation stage [19,20]. BPMN v2.0 falls short at describing those business rules because it does not contain core modeling elements that can accurately depict them.

The aforementioned examples that BPMN v2.0 cannot represent are examples that are based on ubiquitous computing. Ubiquitous computing (frequently referred to as ubicomp) was coined by Mark Weiser around 1988 [66]. Ubicomp denotes the third era of modern computing where one person owns and operates multiple computers (1 person, n computers). The first and the second were respectively mainframe computing (n persons, 1 computer) and personal computing (1 person, 1 computer). While human-computer interactions are typically administered via keyboards and mice as well as display, printers, and speakers, ubicomp adds to these state of the art input technologies such as sensors (e.g., accelerometer, gyrometer, geo-locating sensors), cameras, and microphones [59]. Ubicomp leverages the fact that computers pervade our lives by proposing solutions that bridge the gap between virtual systems and the physical environment (i.e., Internet of Things – IoT [33]).

Ubicomp capabilities may provide the basis for solving many issues in business process management, particularly with respect to process improvement [69–71], compliance [49], and security [68]. When ubicomp elements are included in a business process, we use the term “ubiquitous business process”. A *ubiquitous business process* is a location-independent business process that turns its business environment into a source of data and/or a target of outcome with the least of human interventions [69]. Ubicomp capabilities include, for instance, Automatic Identification and Data Capture (AIDC) [60] (e.g., location-tracking [36], activity-sensing [17]) (which helps to overcome media breaks [26]), context awareness [1], augmented reality [5], sustainability [29], and ambient intelligence [46] that can be included in business processes. In fact, many organizations have already adopted such ubicomp capabilities to cope with the changing business environments and to remain competitive. For instance, UPS¹ overcomes media breaks between the physical and the digital world through bar-code tags to update the status of packages transiting through its logistics

system. Netflix² and YouTube³ use context awareness to recommend the most popular videos in the user's location (i.e., one type of context). TryLive⁴ proposes augmented reality solutions to allow its users to virtually try on apparel. Nest⁵ thermostats support sustainability. Google Now⁶ enables ambient intelligence. However, as BPMN falls behind in representing these business scenarios, the organizations have designed them on their own. Their designs are digressions from the standard, since details touching the process life-cycle such as conformance [54] and compliance [55] remain overlooked. No formal V&V and/or transformation initiative(s) can take place, because the required conditions are not fulfilled. The process personnel in a process-oriented organization are classified into five categories; process owner, process manager, process participant, process analyst and process engineer [16]. Note, process analysts and engineers have direct interaction with BPMN. So, imagine the organization (i.e., process owner) wants to deploy augmented reality in one of its processes. Here, the process analyst has to set the specification for the new process. The process engineer has to make the specification concrete. Then, the analyst should validate its conformance before deploying it for the process participant. In this case, there is higher risk of having more back-and-forth discussions between analysts and engineers because there is no standard medium for them to clearly understand each other. Still, imagine after a certain time, the organization pushes for an improvement of the process and the initial team of analyst/engineer has changed. It would be a challenge for the new one to take over. Even with the old team being unchanged, the situation can also be challenging. This is analogous to writing code and coming back to it after time has passed. Even with comments in the code, it will take some time for the initial programmer to remember the details of the code before being able to weigh in with improvements. This problem can even be exacerbated in the case of a new programmer. Ultimately, without a clear-cut/common specification, the accessibility and use of process diagrams will only exist for the process analysts who created them.

In the scientific literature, we can find several attempts to deploy ubicomp or one of its capacities in business processes. Jung et al. [32] make a proposition of service integration, while Giner et al. [31] take a model driven approach to harmonize the dynamism of business processes and the complexity of ubicomp. While these two approaches attempt to tackle ubicomp as a whole, others focus on specific parts of it. For instance, the authors of [13,14,21] focus specifically on context awareness. Aoumeur et al. [3] and Zhu et al. [72] go even more specific and focus on one aspect of context awareness which is location awareness. A proposition of “*Smart Business Processes*” by means of an RFID integration is discussed in [2].

Although the potential advantages of including ubicomp in business processes are discussed across all the foregoing references, the critical question of how to design ubiquitous business processes remains unanswered. The major obstacle is that BPMN v2.0 cannot represent the ubiquitous computing input technologies. Consequently, it seems paramount to extend it. The present paper presents such an extension that we term “ubiquitous BPMN” (or, in short, uBPMN).

The remainder of this paper is organized as follows: In Section 2, we summarize related work that extended BPMN inside and outside the scope of ubicomp. Building on that, we enumerate the steps needed to coherently extend the notation.

² <https://www.netflix.com/>.

³ <https://www.youtube.com/>.

⁴ <http://www.trylive.com/>.

⁵ <https://nest.com/>.

⁶ <https://www.google.com/landing/now/>.

¹ <http://www.ups.com/>.

Download English Version:

<https://daneshyari.com/en/article/550472>

Download Persian Version:

<https://daneshyari.com/article/550472>

[Daneshyari.com](https://daneshyari.com)