A BPM-Based Approach to Executing Contextualized Actions in Multidomain IoT Systems

Adrian Bazan-Muñoz[1]

[1]UCASE Software Engineering Group, University of Cadiz, Puerto Real, Cadiz, Spain

Abstract

In recent decades, urbanization has put significant pressure on critical urban infrastructure, posing important challenges in terms of sustainability, resource management, and quality of life. Against this backdrop, smart cities have emerged as data-driven, technological ecosystems designed to enhance the efficiency of urban services. The Internet of Things (IoT) plays a pivotal role in this process by enabling real-time monitoring and smart automation via distributed devices. However, while there are many IoT applications in areas such as health, mobility and energy, most of them operate in isolation, using data from a single domain and failing to realize the potential of correlating information across domains. To address this limitation, the following solution was recently proposed as part of an ongoing PhD: (i) a taxonomy that can describe data from different IoT domains and smart environments homogeneously, and (ii) a software architecture that can receive and process these data in real time using a Complex Event Processing (CEP) engine. This architecture enables the detection of situations of interest — defined as meaningful combinations of events and contextual conditions relevant to a given domain — and the execution of contextualized actions. However, despite this progress, the systematic definition of contextualized actions associated with situations of interest is still pending. Furthermore, creating a large number of CEP patterns can significantly increase the system's complexity. To address both issues, we propose using Business Process Models (BPMs), which allow the behavior of the system and the CEP patterns deployed to be graphically represented, and enable the actions to be clearly and flexibly defined depending on the context.

Keywords

Context awareness, Internet of Things, Business Process Model, Complex event processing

1. Motivation

The process of global urbanization has led to accelerated city growth, placing significant pressure on critical infrastructure, including waste management, water consumption, housing, and other urban services. This situation present numerous challenges such as sustainability, quality of life, and resource management. Against this backdrop, the concept of smart city has emerged as a technological response that promotes the use of data-driven solutions to improve the efficiency and quality of urban services.

In this context, the Internet of Things (IoT) has firmly established itself as an essential technology. It allows real-time monitoring via smart devices distributed throughout cities and is capable of generating relevant data for decision-making purposes. This data enables processes to be automated, resources to be optimized, and personalized services to be offered to citizens. The increasing availability of smart environment technologies, as well as reduced communication costs, has favored the development of multiple IoT applications in areas such as health, mobility, energy, and home management.

However, most of these applications are limited in that they only make use of data from a single application domain. This means that they miss a great opportunity to use data from other seemingly unrelated domains. Failing to consider data from other domains means missing the opportunity to improve decision-making and contextualization of IoT applications. This is mainly due to the lack of mechanisms for describing and exchanging data across application domains, which makes it difficult to integrate and correlate data.

23rd International Conference on Business Process Management (BPM 2025):Proceedings of the Best BPM Dissertation Award, Doctoral Consortium, and Demonstrations & Resources Forum co-located with 23rd International Conference on Business Process Management (BPM 2025), Seville, Spain, August 31st to September 5th, 2025

adrian.bazan@uca.es (A. Bazan-Muñoz)
0000-0001-5182-4905 (A. Bazan-Muñoz)

As part of this PhD thesis with the aim of addressing this problem, a solution that includes the following has recently been proposed[1]: (i) A taxonomy for IoT and smart environments was provided, which allows data from different domains to be described in a homogeneous way. (ii) A software architecture was proposed with the aim of receiving and processing information from multiple IoT domains in real time, using a Complex Event Processing (CEP) engine. The CEP engine allows correlating data (so called events) from different domains and detecting situations of interest, combinations of events and conditions that together acquire a meaning relevant to a specific domain. Finally, these detected situations of interest allow the generation of contextualized actions to respond to the detected user needs [2]. For example, consider a smart mobility system that manages a fleet of shared electric vehicles in a city as the main domain. If the system detects that a holiday is being celebrated in a certain area of the city and numerous users are looking for a vehicle in that area at the same time, but the number of available vehicles in the area is low, it could alert the technicians to reassign vehicles to this zone. Alternatively, it could anticipate low car availability, knowing that there is a holiday and the weather is going to be adverse, and therefore increase the number of available vehicles in advance. Furthermore, taking into account contextual data from the energy domain could enable the system to prioritize the reallocation of electric vehicles to nearby charging points with low electricity demand. This would optimize both mobility logistics and the efficient use of urban energy infrastructure. In this way, we can see that CEP improves both vehicle availability and the user experience, as well as contributing to the intelligent use of the city's electric resources.

Despite the fact that the CEP engine detects situations of interest, the definition of contextualized actions is still a research goal pending in this PhD [3]. Additionally, defining a large number of CEP patterns can significantly increase the system's complexity, making maintenance and updates more resource-intensive than is advisable. To address these issues, the use of Business Process Models (BPM) is proposed. These graphical representations can model both the system's general behavior and the CEP event patterns deployed in the engine, as well as define the contextualized actions associated with the event patterns. In this way, BPM can be used to model contextualized actions for each user of the system, which should be taken when a situation of interest is detected, based on the user's context.

The rest of the document is structured as follows. Section 2 presents related work. Section 3 describes the research methodology employed in this phase of the PhD. Section 4 then explains the solution envisioned, detailing the envisaged extension to the architecture and presenting a case study. Finally, Section 5 presents the conclusions.

2. Related Work

BPM has been explored on different occasions in the context of IoT applications and smart environments. Some proposals have demonstrated how BPM can be employed to define processes within context-aware systems. In the healthcare domain, for example, Ruiz-Fernandez et al. [4] proposed a BPM model for managing the entire process of monitoring patients with hypertension. In the context of automated canteens, Fedeli et al. [5] suggested using BPM to integrate devices such as proximity sensors, scanners, and pressure elements. In this case, these devices were used as part of the BPM process to activate actuators, thereby contributing to the automation of the system. Valderas et al. [6] proposed modelling an intelligent logistics center using BPM. This process definition enables the logistics center to operate autonomously with minimal human intervention. While these proposals demonstrate the potential of BPM for process management in the IoT and smart environments, they mostly focus on defining processes that respond to events within the domain itself. They do not consider data from other domains or the management of contextualized actions in the IoT and smart environments.

More recent proposals have explored the possibility of combining BPM and CEP engines. Ruhkamp et al. [7] proposed that integrating both technologies would enable situations of interest to be identified in real time and would allow the BPM model to define how human operators should intervene when an error occurs in the system. Building on this idea, approaches have emerged that use BPM processes to model CEP engine operations [8], or even make use of data mining for this purpose [9]. However, these solutions tend to focus on waiting for simple events or the activation of patterns representing interaction with the CEP engine at a higher abstraction level, but without representing the CEP patterns themselves, which would lead to a better understanding of the system.

This analysis shows that, although there is growing interest in integrating CEP and BPM in IoT and smart environments, the representation of CEP patterns in BPM models has yet to be addressed. Nor has a solution been found that enables contextualized actions to be defined in response to different situations of interest that are detected and correlated in real time from multiple contextual domains. In this sense, we believe that the integration of CEP and BPM could help define contextualized situations and actions in IoT domains, and this is one of the gaps we aim to address in this doctoral thesis.

3. Research Methodology

In this section, the methodology proposed by Hevner et al. [10], and further formalized by Wieringa [11] is adopted to guide the development and validation of integrating CEP and BPM in smart environments. Thus, we have two main artifacts in our proposal. A software architecture enabling CEP-BPM integration to support contextualized actions, and a modelling approach to represent CEP patterns within BPM models. The research questions (RQ) are as follows:

RQ1. How can a CEP engine be integrated with BPM in order to define contextualized actions for situations of interest?

This research question focuses on understanding how BPM models can evolve from simple diagrams that describe processes into active components of the system that can react to situations by identifying them through the integration of the CEP engine.

To this end, we aim to:

1. Integrate the CEP engine with BPM by extending the previously proposed software architecture.
2. Implement a case study to illustrate how contextualized actions are triggered by situations of interest detected in the CEP engine through the BPM model.
3. Analyse the case study to check whether the BPM model correctly represents how the different actors in the system behave and interact with each other.
4. Evaluate the results obtained in relation to the set goal.

RQ2.Can BPM models be used to represent CEP patterns in detail, thus making them easier to maintain, understand and evolve?

This research question aims to analyze the potential of BPM models as a tool for representing and structuring CEP event patterns, with the ultimate goal of improving understanding, maintaining and adapting to changing contexts. To this end:

1. Analyze the components of a CEP pattern and how they can be represented in BPM. In other words, we aim to analyze how simple events, time windows, logical conditions and other elements from the event processing language can be represented in BPM.
2. Define a case study consisting of a series of CEP patterns that depend on each other, such that the output of one pattern becomes the input of the next. Furthermore, implement this series of dependent patterns in BPM to demonstrate how these models can represent the CEP engine behavior.
3. Extend the case study to present the information in accordance with the architecture. This should represent all actors, the CEP engine and its defined CEP patterns, and the contextualized actions in response to situations of interest detected in the patterns.
4. Test with different users with CEP knowledge to validate that the representation of CEP patterns in BPM improves the system's maintainability and comprehensibility.

In summary, following the previously mentioned methodology, the research questions seek to validate the feasibility of integrating CEP and BPM, and to explore the benefits in terms of maintainability and evolution of defining CEP patterns in BPM. Through the implementation of case studies and the analysis of the results, it is expected to obtain evidence that will inform the definition of contextualized actions in IoT and smart environments. These results will contribute to the final phase of this thesis, achieving its objectives.

4. Solution envisioned

According to the stated RQ, we aim to extend the previously proposed context-aware software architecture by using BPM to define CEP patterns and associated contextualized actions. Figure 1 sketches an initial representation of the proposed extension. The software architecture's information flows would be as follows:

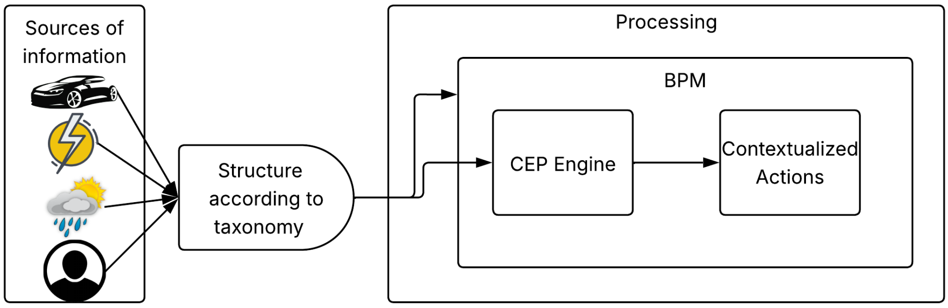

Figure 1: Envisioned Software Architecture

- Firstly, the events provided by the IoT scenario arrive at the BPM model via a messaging broker structured according to the previously defined taxonomy.
- In parallel, the events arrive at the CEP engine, where they are processed to detect situations of interest. These CEP patterns are defined in the BPM model.
- Finally, each situation of interest that is detected has an associated contextualized action that is also defined in the BPM model. These actions are executed when the relevant situation is detected, allowing the system to respond to the user's needs.

To illustrate the proposal, a brief case study modelled in BPM will be presented. This will consider a smart mobility system that manages a fleet of shared electric vehicles in a city as the main domain. The main objectives are to improve vehicle availability, enhance the user experience and reduce energy consumption by making use of contextual data from the smart city. Although we envision the architecture being implemented entirely within BPM, with the CEP engine execution invoked by the Business Process Management Suite (BPMS) engine, it would also be possible to implement the CEP engine externally, communicating with BPM to execute detected contextualized actions. The operation would proceed as follows (see Figure 2):

- First, the BPM models which represent the CEP patterns in the CEP engine are defined using Business Process Model and Notation (BPMN). The BPM model will specify the information to be received by each event pattern, which should be structured in accordance with the taxonomy, as well as the actor that generates such information. In this case, the pattern will receive contextual information that a holiday is being celebrated in a certain area of the city and that numerous users are looking for a vehicle in that zone at the same time. Alternatively to users, the pattern may receive information that the weather conditions are adverse. Subsequently, information on the number of vehicles available in each zone will be received. All this information will be received and analyzed by an event pattern called high demand alert. Finally, electricity data will be received from various recharging points in the area.

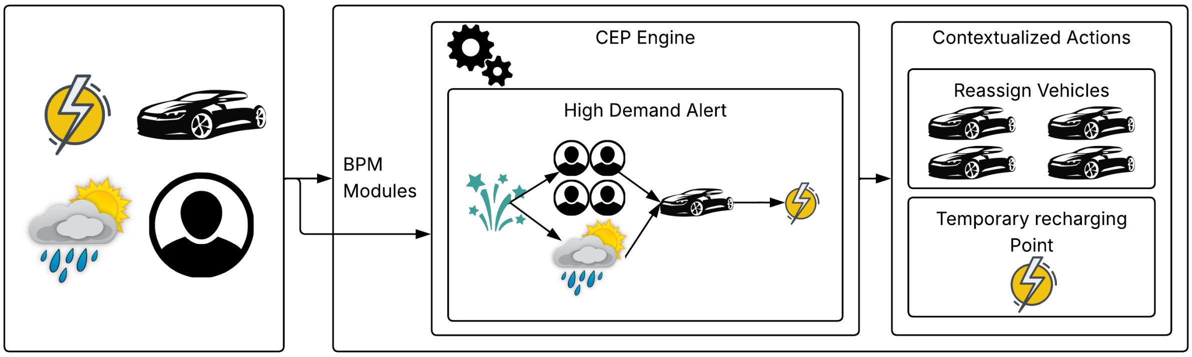

Figure 2: Case study

- Secondly, once all necessary information has been received, the CEP engine will evaluate the defined clauses. A complex event will be generated, as will an alert as there is too much demand for too little availability in the area or when weather conditions are adverse. This situation of interest will be sent to the BPM module for contextualized actions.
- Finally, once a situation of interest associated with a stakeholder has been detected, the contextualized action defined for it in the BPM will be executed. In this case, technicians will be contacted to reassign vehicles to the affected area as soon as possible and direct them to the recharging points with the lowest electricity demand. In addition, if the recharging points in the area were overloaded or not working, another action would be triggered to alert technicians to the need to set up a temporary recharging point.

5. Conclusion

So far, we have proposed a PhD project focusing on developing a system that can integrate and correlate data from different application domains. The aim is to enrich the context of intelligent applications and improve decision-making. This integration enables situations of interest to be detected that would otherwise remain undetected. As part of the natural evolution of this work, we are now focused on defining response mechanisms to such situations and facilitating the definition of CEP patterns. To this end, we propose a BPM-based solution, which enables the modelling of interactions between system actors, the CEP engine, and contextualized actions to be executed according to detected situations. This new proposal is an extension of the initial approach presented in [1], which incorporates the definition and management of contextualized actions. We expect to obtain results shortly that will enable us to consolidate this solution and explore its applicability in other domains. This will enable aspects such as the proposal's usability, adaptability, and effectiveness to be evaluated in the future by testing it with developers and IoT domain experts.

Acknowledgments

This publication is part of the I+D+i grant PID2021-122215NB- C33 (AwESOMe) funded by MICIU/AEI/10.13039/501100011033 and by ERDF/EU and by the pre-doctoral research staff training contract 2022-006/PU/EPIF-FPI-CT/CP of the University of Cadiz

Declaration on Generative AI

The author has not employed any Generative AI tools.

References

[1] A. Bazan-Muñoz, G. Ortiz, J. C. Augusto, A. Garcia-de Prado, Taxonomy and software architecture for real-time context-aware collaborative smart environments, Internet of Things 26 (2024) 101160. URL: https://www.sciencedirect.com/science/article/pii/S254266052400101X. doi:10.1016/j.iot.2024.101160.

[2] M. Daum, M. Götz, J. Domaschka, Integrating CEP and BPM: how CEP realizes functional requirements of BPM applications (industry article), in: Proceedings of the 6th ACM International Conference on Distributed Event-Based Systems, DEBS '12, Association for Computing Machinery, New York, NY, USA, 2012, pp. 157–166. URL: https://doi.org/10.1145/2335484.2335503. doi:10.1145/2335484.2335503.

[3] P. Soffer, A. Hinze, A. Koschmider, H. Ziekow, C. Di Ciccio, B. Koldehofe, O. Kopp, A. Jacobsen, J. Sürmeli, W. Song, From event streams to process models and back: Challenges and opportunities, Information Systems 81 (2019) 181–200. URL: https://www.sciencedirect.com/science/article/pii/S0306437917300145. doi:10.1016/j.is.2017.11.002.

[4] D. Ruiz-Fernández, D. Marcos-Jorquera, V. Gilart-Iglesias, V. Vives-Boix, J. Ramírez-Navarro, Empowerment of Patients with Hypertension through BPM, IoT and Remote Sensing, Sensors 17 (2017) 2273. URL: https://www.mdpi.com/1424-8220/17/10/2273. doi:10.3390/s17102273, number: 10 Publisher: Multidisciplinary Digital Publishing Institute.

[5] A. Fedeli, F. Fornari, A. Polini, B. Re, V. Torres, P. Valderas, FloBP: a model-driven approach for developing and executing IoT-enhanced business processes, Softw Syst Model 23 (2024) 1217–1246. URL: https://doi.org/10.1007/s10270-024-01150-8. doi:10.1007/s10270-024-01150-8.

[6] P. Valderas, V. Torres, E. Serral, Modelling and executing IoT-enhanced business processes through BPMN and microservices, Journal of Systems and Software 184 (2022) 111139. URL: https://www.sciencedirect.com/science/article/pii/S0164121221002363. doi:10.1016/j.jss.2021.111139.

[7] N. Ruhkamp, S. Schönig, Execution of Multi-Perspective Declarative Process Models Using Complex Event Processing, Business Information Systems (2021) 95–104. URL: https://www.tib-op.org/ojs/index.php/bis/article/view/51. doi:10.52825/bis.v1i.51.

[8] C. Janiesch Jörn Kuhlenkamp, Enhancing business process execution with a context engine, 2021. URL: https://ui.adsabs.harvard.edu/abs/2021arXiv211004061J. doi:10.48550/arXiv.2110.04061, aDS Bibcode: 2021arXiv211004061J.

[9] I. M. A. Helal, A. Awad, Online correlation for unlabeled process events: A flexible CEP-based approach, Information Systems 108 (2022) 102031. URL: https://www.sciencedirect.com/science/article/pii/S0306437922000333. doi:10.1016/j.is.2022.102031.

[10] A. Hevner, A. R, S. March, S. T, Park, J. Park, Ram, Sudha, Design Science in Information Systems Research, Management Information Systems Quarterly 28 (2004) 75.

[11] R. Wieringa, Design Science Methodology for Information Systems and Software Engineering, 2014. doi:10.1007/978-3-662-43839-8, pages: 332.