



Towards a Formal Context-Aware Workflow Model for Ambient Environment

Roumeissa Khennaoui^(✉) and Nabil Belala^(✉)

MISC Laboratory, University of Constantine 2 Abdelhamid Mehri,
Constantine, Algeria
{roumeissa.khennaoui,nabil.belala}@univ-constantine2.dz

Abstract. Ambient systems owns some particular characteristics that makes their context awareness a sincere problem; they are composed of heterogeneous distributed devices, some of these devices may appear and disappear during operations. In addition, users interacting in these systems are themselves dynamic. Therefore, context-aware workflow management allows workflows to adapt dynamically according to the environment changes. Context information are complex and diverse which makes the modeling the key issue. This paper presents an approach to model context-aware workflows. First, we describe the workflow using Ag-LOTOS. Then, based on this description, we build the contextual planning system *CPSw* that allows the presentation of the context at each activity state.

Keywords: Modeling of physical and conceptual information in smart environments · Context awareness · Ambient intelligence · Formal description · Workflow

1 Introduction

In order to meet the needs of everyday life, systems are becoming more and more complex, which leads to seek to give more atomicity and initiative to the different software modules. To respond to this technological evolution, ambient intelligence [1] is a new paradigm of distributed systems where the environment is aware of the user's needs and find a way to fulfill that need to improve the quality of people's life.

Due to the extreme mobility of users, ubiquitous software [2] run in a highly dynamic and varying environment. Therefore, context awareness [3] and context adaptation [4] are some important aspects for pervasive software that have to be aware of the context's changes, and dynamically adjust their execution [4]. Context-aware workflow [5] is an interesting field that allows workflows to adapt dynamically to the context changes in ubiquitous environment.

To achieve this goal, no many results have been accomplished in workflow's context modelling. [6] proposes a context-aware workflow management system

(WFMS) for navigation applications in ubiquitous computing. In [7], a dynamic context-aware access control for pervasive computing in enterprise environment is proposed. However, [8,9] allow users to model their daily activities in the form of workflow adaptable to context information. [10] proposes an approach to build a flexible model to adapt business process based on context. In [11], both the conceptual model and the workflow model are defined based on OWL.

Considering that the ambient systems manage our daily life such as smart hospital, smart home, robots, etc., errors are critical regarding human life. Many WFMS tools [12,13] exist and allow the modelling and verification of workflows. However, mathematical approaches are proved to be more effective [14]. [15] describes the workflow patterns in the formal specification language LOTOS [16]. In addition, [17] proposes an approach to specify and verify the service composition using LOTOS.

In this paper, we describe at first the workflow using Ag-LOTOS [18], a formal specification model based on LOTOS. Ag-LOTOS is a formal technique based on process algebra that allows to formally describe the workflow and to verify properties on the model. Then, the contextual planning system of the workflow (CPSw) [19] is built based on the semantics of Ag-LOTOS constrained by contextual information. Unlike the previous work, our approach allows a formal description of the context as *pre*- and *post*-condition in each state of all the possible traces and adjust the changes dynamically. The proposed model can be used in the verification process to check some contextual properties.

2 The Context-Aware Workflow Model

2.1 Ag-LOTOS for Workflows

Business Process Management (BPM) has been defined by van der Alast as “a way to support business processes using methods, technique and software to model, execute, control and analyze operational processes involving humans, organizations, applications, documents or any other source of information” [20]. However, a workflow can be defined as “business process automation during which documents, information or tasks are passed from one participant to another according to a set of process rules” [20]. A workflow pattern [21] represents the abstraction of most frequent activities sequence, and are composed when specifying new workflows.

In this section, we aim to use Ag-LOTOS [18] to improve workflow specification by including contextual information to each state of the workflow model, and by modeling the ambient characteristics such as communication and mobility that cause the dynamic changes of the context.

Similar to LOTOS [16], Ag-LOTOS concurrency allows the modeling of parallel activities. The Ag-LOTOS subsystem support allows the composition of workflow elements. Since Ag LOTOS is derived from LOTOS, we follow the procedure of mapping of the workflow patterns to LOTOS notation applied in [14] and cite6 to give the suitable definition of each operator in the workflow context. To specify activities in details, we can simply model them with Ag-LOTOS sub-processes (hierarchy of processes). Note that Ag-LOTOS processes are used

to model activities in the workflow. However, the process in the workflow context indicates a set of activities.

Ag-LOTOS expressions are written by composing actions through the LOTOS operators.

The syntax is defined as follows [18]:

$$\begin{aligned}
 P &::= E \\
 E &::= \textit{exit}|\textit{stop} \\
 a; E | E \odot E \quad (a \in \partial) \\
 &| \textit{hide } L \textit{ in } E \\
 \mathcal{H} &::= \textit{move}(l) \quad (\mathcal{H} \subset \partial, l \in \ominus) \\
 &| x!(v)|x?(v) \quad (x \in \mathcal{U}, v \in \mathcal{M}) \\
 \odot &::= \{[L], ||, \gg, [], \parallel, >\}
 \end{aligned}$$

Where ∂ is a finite set of observable actions, L is a subset of ∂ and $\mathcal{H} \subset \partial$ is the set of ambient intelligence primitives, which represent the mobility and the communication. \ominus is the finite set of spatial localities of the pervasive environment, \mathcal{U} is a finite or infinite set of users, with which the user can communicate, and \mathcal{M} is the set of messages that can be sent or received.

An essential component of a process definition is its behavior expression E . A behavior expression is built by applying an operator, e.g., \gg , to other behavior expressions. A behavior expression may also include instantiations of other processes.

Termination. In Ag-LOTOS, the termination is represented via the operator *stop* which indicates the inaction while the *exit* operator expresses the successful termination.

Fail. In $A = \textit{fail}$, *fail* represents the fact that the execution of an activity A fails because of the dynamic context of the workflow.

Prefix. The operator ‘;’ is used to prefix a behavior expression with an action to produce a new one. Note that actions are the elementary units executed by activities.

Hiding. *hide* is used to express the discriminator pattern (similar to LOTOS). An external gate is used to invoke the subprocess that enables the activity. This gate is hidden inside the discriminator to avoid any external synchronization (see [14, 15] for further details).

Respectively, the set \odot represents the standard LOTOS operators.

Sequence. The sequential composition operator \gg is used to represent the sequence pattern.

Cycle. A loop in a process allows the repetitive execution of activities, $P ::= E \gg P$.

Choice. $A [] B$, activity A or B will be chosen.

Disabling. During the activity execution, it is possible to indicate its failure with the disabling operator $[>]$. $A [> B$ means activity A may be disabled by activity B which interrupts the main flow and uses *stop* instead of *exit*.

Parallelism (general case). $A [[L]] B$ means if the process (activity A) is ready to execute some action at one of the synchronization gates, it is forced, in the absence of alternative actions, to wait until the process (activity B) offers the same action.

Full Synchronization. $A \parallel B$ means that if $L = \partial$, the two composed activities are forced to execute in complete synchronicity.

Pure Interleaving. If $L = \emptyset$, the absence of synchronization leads to the absence of interaction points among processes, this is achieved through the interleaving operator $'|||'$.

2.2 Contextual Planning System of the Workflow

In order to illustrate the concept of the formal design of workflows with the contextual information, the contextual planning system is built from an Ag-LOTOS specification using the rules in Table 1.

Table 1. The semantic rules.

Action:	$\frac{ws \xrightarrow{a} ws' \quad a \in Act}{(ws, l) \xrightarrow{a} (ws', l)}$
Mobility:	$\frac{ws \xrightarrow{move(l')} ws' \quad (l \neq l')}{(ws, l) \xrightarrow{move(l')} (ws', l')}$
Communication:	$(a) \frac{ws \xrightarrow{x!(u)} ws' \quad (u \in \mathcal{U})}{(ws, l) \xrightarrow{x!(u)} (ws', l)} \quad (b) \frac{ws \xrightarrow{x?(u)} ws' \quad (u \in \mathcal{U})}{(ws, l) \xrightarrow{x?(u)} (ws', l)}$

The Contextual Planning System of the Workflow (CPSw) based on CPS [22] takes into account two types of information: workflow planning state ws and locality l . Table 1 shows the operational semantic rules that define the possible planning state changes for the workflow. From an initial planning state (ws_0, l) , we apply these rules to produce the CPSw. The contextual planning system CPSw is a labeled Kripke structure (S, s_0, Tr, L) where S is the set of contextual planning workflow states, $s_0 = (ws_0, l) \in S$ is the initial planning state of the workflow, $Tr \subseteq S \times \partial \cup \{T\} \times S$ is the set of transitions which are denoted $s \xrightarrow{a} s'$, and $L : S \rightarrow \ominus$ is the location labeling function.

3 Case Study

In this paper, we target on context-aware workflow models for ubiquitous company. Let there be an enterprise with several helpdesk employees associated with smart badges that provide the system with spatial information at each moment.

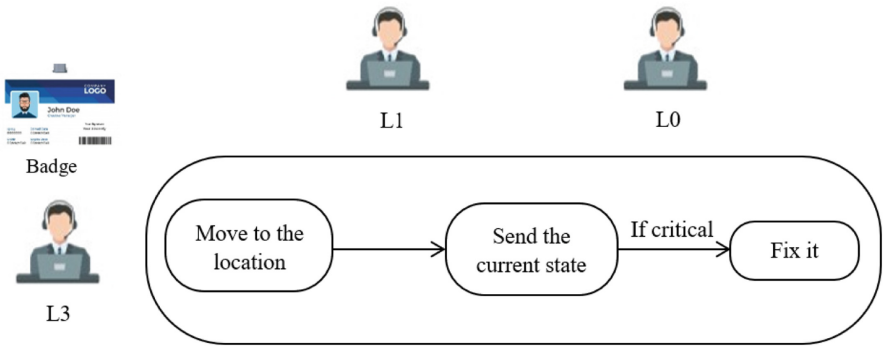


Fig. 1. The scenario that illustrate the case study.

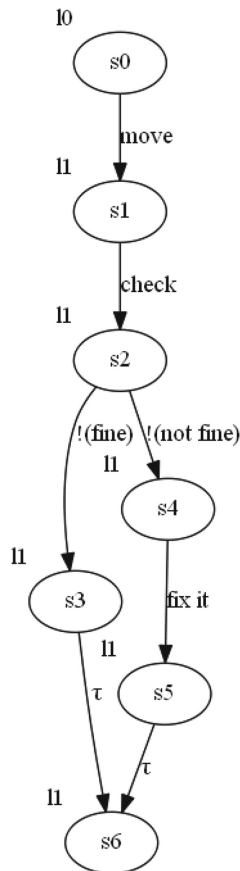


Fig. 2. The CPSw corresponding tot the case study.

The scenario illustrated in Fig. 1 is highly context dependent, especially in the following way:

If the system detects a problem on the switch in datacenter location, he sends a request to the closest helpdesk. This one has to move to the location of the switch and send the current state by email to the management system:

State 1 `fine`, indicate that there is no critical problem.

State 2 `not_fine`, in case of critical one which need to be fixed.

The corresponding CPSw to the scenario is illustrated in Fig. 2. It is built from the initial Ag-LOTOS description:

```
move(l1); check;
  (x!(fine); exit [] x!(not_fine); fix_it(l1); exit
```

4 Conclusion

Workflow systems are currently used by many organizations including health care, automation and finance. Context awareness is the ability for workflows to react to the changing situations. In this paper, we introduced a context-aware workflow model, the CPSw, that presents all the possible evolutions of workflow's activities constrained by the contextual information. CPSw is constructed formally based on Ag-LOTOS description giving the set of activities.

We learned that using Ag-LOTOS to describe workflow activities is a promising approach. Mainly, because it allows a formal description of the current context in each state as *pre*- and *post*-conditions, and dynamically adjusts the modifications. Furthermore, it allows the verification and validation of the model.

The proposed model can be used in the verification process to verify certain contextual properties. For future works, we aim to consider different types of context information such as the time.

References

1. Remagnino, P., Foresti, G.: Ambient intelligence: a new multidisciplinary paradigm. *IEEE Trans. Syst. Man Cybern. - Part A: Syst. Hum.* **35**(1), 1–6 (2004)
2. Ramos, C.: Ambient intelligence a state of the art from artificial intelligence perspective. In: Neves, J., Santos, M.F., Machado, J.M. (eds.) *EPIA 2007*. LNCS, vol. 4874, pp. 285–295. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-77002-2_24
3. Abowd, D., Dey, A.K., Orr, R., et al.: Context-awareness in wearable and ubiquitous computing. *Virtual Reality* **3**(3), 200–211 (1998)
4. Smachat, S., Ling, S., Indrawan, M.: A survey on context-aware workflow adaptations. In: *Proceedings of the 6th International Conference on Advances in Mobile Computing and Multimedia*, pp. 414–417 (2008)
5. Wieland, M., Kopp, O., Nicklas, D., Leymann, F.: Towards context-aware workflows. In: *CAiSE07 Proceedings of the Workshops and Doctoral Consortium*, vol. 2, no. S25, p. 15 (2007)

6. Tang, F., Guo, M., Dong, M., Li, M., Guan, H.: Towards context-aware workflow management for ubiquitous computing. In: 2008 International Conference on Embedded Software and Systems, pp. 221–228. IEEE (2008)
7. Zhu, Z., Xu, R.: A context-aware access control model for pervasive computing in enterprise environments. In: 2008 4th International Conference on Wireless Communications, Networking and Mobile Computing, pp. 1–6. IEEE (2008)
8. Avenoglu, B., Eren, P.E.: A context-aware and workflow-based framework for pervasive environments. *J. Amb. Intell. Hum. Comput.* **10**(1), 215–237 (2019)
9. Carolis, B.D., Ferilli, S., Redavid, D.: Incremental learning of daily routines as workflows in a smart home environment. *ACM Trans. Interact. Intell. Syst. (TiiS)* **4**(4), 1–23 (2015)
10. Da Cunha Mattos, T., Santoro, F.M., Revoredo, K., et al.: A formal representation for context-aware business processes. *Comput. Ind.* **65**(8), 1193–1214 (2014)
11. Wang, P., Li, H., Zhang, B.: Context-aware workflow modeling approach using OWL. In: The 26th Chinese Control and Decision Conference (2014 CCDC), pp. 4161–4165. IEEE (2014)
12. Salimifard, K., Wright, M.: Petri net-based modelling of workflow systems: an overview. *Eur. J. Oper. Res.* **134**(3), 664–676 (2001)
13. Sadiq, S., Orlowska, M., Sadiq, W., et al.: Data flow and validation in workflow modelling. In: Proceedings of the 15th Australasian Database Conference, vol. 27, pp. 207–214 (2004)
14. Carchiolo, V., Longheu, A., Malgeri, M.: Using LOTOS in workflow specification. In: ICEIS, no. 3, pp. 364–369 (2003)
15. Takecian, P. L., Ferreira, J. E., Malkowski, S., Pu, C.: Using LOTOS for rigorous specifications of workflow patterns. In: 6th International Conference on Collaborative Computing: Networking, Applications and Worksharing (CollaborateCom 2010), pp. 1–7. IEEE (2010)
16. Bolognesi, T., Brinksma, E.: Introduction to the ISO specification language LOTOS. *Comput. Netw. ISDN Syst.* **14**(1), 25–59 (1987)
17. Dumez, C., Bakhouya, M., Gaber, J., Wack, M.: Formal specification and verification of service composition using LOTOS. In: Proceedings of the 7th ACM International Conference on Pervasive Services (2010)
18. Chaouche, A.C., Seghrouchni, A.E.F., Ilié, J.M.: A formal approach for contextual planning management: application to smart campus environment. In: Bazzan, A., Pichara, K. (eds.) *IBERAMIA 2014*. LNCS, vol. 8864, pp. 791–803. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-12027-0_64
19. Chaouche, A.C., Seghrouchni, A.E.F., Ilié, J.-M., et al.: A higher-order agent model with contextual planning management for ambient systems. In: Kowalczyk, R., Nguyen, N. (eds.) *Transactions on Computational Collective Intelligence XVI*, pp. 146–169. LNCS, vol. 8780. Springer, Heidelberg (2014). https://doi.org/10.1007/978-3-662-44871-7_6
20. van der Aalst, W., Ter Hofstede, A., Weske, M.: Business process management: A survey. In: van der Aalst, W.M.P., Weske, M. (eds.) *BPM 2003*. LNCS, vol. 2678, pp. 1–12. Springer, Heidelberg (2003). https://doi.org/10.1007/3-540-44895-0_1
21. van der Aalst, W., Ter Hofstede, A., Kiepuszewski, B., et al.: Workflow patterns. *Distrib. Parallel Databases* **14**(1), 5–51 (2003)
22. Chaouche, A.C., Seghrouchni, A.E.F., Ilié, J.-M., et al.: From intentions to plans: a contextual planning guidance. In: Camacho, D., Braubach, L., Venticinque, S., Badica, C. (eds.) *Intelligent Distributed Computing VIII. Studies in Computational Intelligence*, vol. 570, pp. 403–413. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-10422-5_42

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.

