Context-aware Workflow Model for Supporting Composite Workflows

Jong-sun CHOI[1], Jae-young CHOI[1], Yong-yun CHO[2]
(1. *School of Computing, Soongsil University, Seoul* 156-743, *Korea*;
2. *Division of Information and Communication, Sunchon National University, Jeonnam* 540-742, *Korea*)

Abstract – In recent years, several researchers have applied workflow technologies for service automation on ubiquitous computing environments. However, most context-aware workflows do not offer a method to compose several workflows in order to get more large-scale or complicated workflow. They only provide a simple workflow model, not a composite workflow model. In this paper, the autorhs propose a context-aware workflow model to support composite workflows by expanding the patterns of the existing context-aware workflows, which support the basic workflow patterns. The suggested workflow model offers composite workflow patterns for a context-aware workflow, which consists of various flow patterns, such as simple, split, parallel flows, and subflow. With the suggested model, the model can easily reuse few of existing workflows to make a new workflow. As a result, it can save the development efforts and time of context-aware workflows and increase the workflow reusability. Therefore, the suggested model is expected to make it easy to develop applications related to context-aware workflow services on ubiquitous computing environments.

Key words – *ubiquitous computing*; *context-aware workflow*; *workflow model*; *multiple workflows*

Manuscript Number: 1674-8042(2010)02-0161-05

dio: 10.3969/j.issn.1674-8042.2010.02.15

1 Introduction

As the workflow technologies have been well applied to the distributed computing environments to provide reliable IT services based on business process standardization, these technologies have been successfully commercialized. Ubiquitous computing has been a top of conversation in the IT area, and the focus of workflow technologies has been moved from distributed computing environments to ubiquitous computing environments. In recent years, some studies have applied workflow models to ubiquitous computing environments in order to provide users with automated services based on situational information. Also there are many studies trying to apply context-aware technologies on ubiquitous computing environments to existing workflow models[1-5]. Automated services for the users on ubiquitous computing environments are usually expected to be fit in the surrounding situation. Workflow systems in ubiquitous computing environments would be able to provide context-aware services to users according to a workflow service scenario, in which various situational information in the surroundings of users are described as service execution conditions[4]. Therefore, the context-aware workflow models on ubiquitous computing environments would express not only situational information, but also composite context-aware services. In addition, it would be able to support multiple workflow services for many users. The studies of Ref. [6] and Ref. [7] mentioned about the importance of multiple workflows and composite workflows.

However, most existing context-aware workflow models have difficulties in composing a new workflow using several workflows because the studies to apply context-aware technologies to workflow models are still in early stage and most of them provide simple workflow services to users. This paper introduces a new context-aware workflow service model to find a solution for that problem. Therefore, our model can make various services be provided to many users. With the suggested model, context-aware service developers can reuse and composite individual workflows to make an integrated and complicated workflow. These developers are also able to raise efficiency in development and workflow reusability.

2 Related works

Workflow techniques that have been supported by universal computer companies, such as Microsoft, IBM, Bea, Oracle, etc. have been constantly developing into workflow languages, such as XLANG, WSFL, and WS-BPEL. The BPEL4WS[8] is a workflow language based on Web services technologies and ncludes the advantages of

* **Received**: 2010-03-25
Project supported: This research was supported by the The Ministry of Knowledge Economy, Korea, the ITRC (Information Technology Research Center) support program (IITA-2009-(C1090-0902-0007))
Corresponding author: Jong-sun CHOI(jschoi@ss.ssu.ac.kr)

both WSFL[9] and XLANG[10] technologies. XLANG is one of the initial workflow languages using a directed graph pproach, and WSFL is a block-structured workflow language used to describe Web services. The workflow languages are able to use results of previous Web services or event information transferred through XPath, XLink, and Xpoint[11] as the current service's transition conditions. Situational information occurred in ubiquitous computing environments is composed of the information, such as user profile, user's location, and time.

However, Ref. [8] ~ Ref. [10] have difficulties in supporting a workflow model to provide context-aware workflow services. Workflow services in ubiquitous computing environments must be developed based on context for situational information, which can be obtained by both users and their surroundings. A context-aware workflow system to execute these services also must use the information as the transition conditions of services. FollowMe[12], an OSGi framework that unifies a workflow-based application model and a context model based on ontology, is one of the implementation models of context-aware services in ubiquitous computing environments. It also uses a scenario-based workflow model to handle the user's demands of services in various service domains. To apply a workflow model to describe context-aware applications, FollowMe introduces Compact Process Definition Language (CPDL), which according to WfMC standards, defines workflow processes by tailoring and recasting XML Process Definition Language (XPDL). Nevertheless, FollowMe needs to supplement something in the design of CPDL to provide many users with context-aware services, because it focuses only on a specific application model based on a workflow. Even if FollowMe considers user's situational information or contexts as workflow service execution conditions, it does not offer an explicit method to independently handle workflow service scenarios.

3 Workflow model for multiple context-taware services

Workflow technologies for distributed and business process environments provide various patterns for service automation. However, the existing workflow technologies adopted on ubiquitous computing environments use limited workflow patterns, such as sequence, exclusive choice, synchronization, parallel split, simple merge, and so on. The proposed workflow models in this paper is able to support more advanced workflow patterns than existing basic models to provide multiple context-aware workflow services to users. Our model also provides the composite workflow patterns, which are expanded or composed of the basic workflow patterns[13]. The models based on these patterns are described in the Subsection A to E of Section 3. Simple workflow model shown in Fig. 1 is basic or simple workflow type, and each model shown in Fig. 1, 2 and 3 is a composite workflow type. These composite types are composed of two or more simple workflow types.

3.1 Simple workflow

Simple workflow is a deterministic flow that various service nodes are connected in a sequence. It is based on a sequence pattern, which is one of the basic workflow patterns.

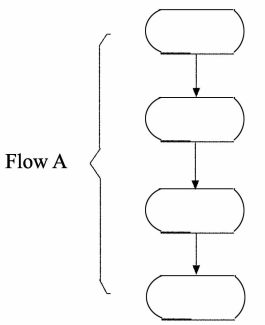

Fig. 1 Simple workflow

3.2 Sub-workflow

Sub-workflow is based on both exclusive choice and synchronization pattern. It means, as shown in Fig. 2, another workflow is attached on one of the nodes on Flow B. Sub-workflow is used to build more complicated workflow services and is good for the improvement of reusability of existing workflows. It is similar to the function call in programming language. For example, Flow C shown in Fig. 2 is the sub-workflow of Flow B. This means that functions in routines call other subroutines.

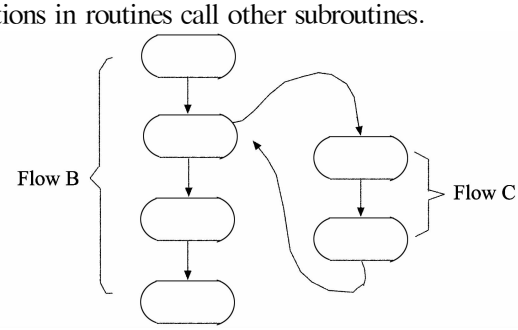

Fig. 2 A subworkflow

3.3 Parallel workflow

Parallel workflow can be used to build more complicated workflow services by using the basic workflow patterns such as the parallel split, synchronization, and simple merge pattern. A parallel workflow can be not-deterministic, even though the flows that included with the parallel workflow are deterministic. Using parallel workflow model, workflow system is able to manage two or more workflows as one larger-scale workflow. Fig. 3 shows an example of the parallel workflow. Flow D con-

sists of few flows and Flow E and F are parts of the Flow D. Transition from node A to Node B and C splits a workflow into two flows in the parallel workflow. And then, however the parallel workflows do not mean that all parallel flows should be joined.

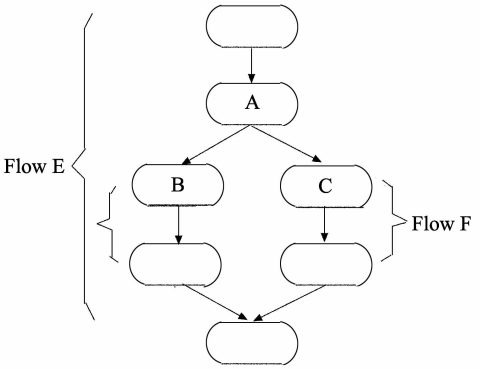

Fig. 3 Parallel workflow

3.4 Multiple workflows

Multiple workflows mean that two or more services are simultaneously provided to many users. Therefore, multiple workflows should support many simple workflows and composite workflows with many users at once. As you can see from Fig. 1, 2 and 3, in a same workflow system, each user can use any kind of workflow services at the same time.

3.5 A multiple context-aware workflow model

Particularly, this paper introduces a composite context-aware workflow model in ubiquitous computing environments. This model is able to reuse or composite existing simple service flows to create new service flows. It can include models with simple or composite workflow types. The composite types are expanded from the basic workflow patterns. Fig. 4 shows the conceptual architecture of the suggested composite context-aware workflow model.

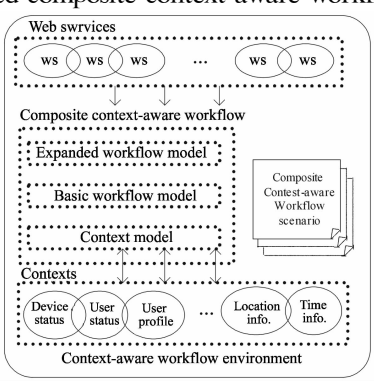

Fig. 4 A conceptual architecture of the suggested multiple context-aware workflow model

In Fig. 4, the context model can be used to select services suitable for users with context information such as user profile, device status, location, and time. And context information is able to be described by using the context model included in composite workflow model. The basic workflow model is based on basic workflow patterns and includes context model. Also it can be used to build simple workflows. And then, expanded workflow model build on the context model and basic workflow model. This model can be used to build new larger-scaled composite workflows. All of these models can be used to control the flows of workflow services and to support multiple workflows to simultaneously provide various services to many users. Therefore, service developers are able to develop various workflow services with these models.

There are various service domains in ubiquitous computing environments. Thus, various context-aware services should be provided with users. For that, service developers are able to describe many composite context-aware workflow scenario documents, as shown in Fig. 4. Each document can be used independently for one user, or it can be used to build new multiple workflow services for many users. Also service developers can express composite workflow services by reusing the existing workflow documents.

4 Experiment and results

In this section we will take a closer look at the experiment for application of the suggested context-aware workflow model. We are going to apply the suggested multiple context-aware workflow model to two scenarios in a home service domain. The first scenario is a home-care service.

1) Mr. Smith's house has been empty for a month. A home-care service system manages the temperature and humidity of his house.

2) When he enters the house, the home-care system increases the temperature and closes the window, which has been opened for the ventilation of rooms.

3) He stops in front of refrigerator to drink water, and then home-care system displays the warning message: "Some food passed the expiration date."

4) After he cleans the refrigerator, the recommended menu for his dinner will be displayed.

5) He selects a menu and prints the recipe. When he starts cooking, the ventilation fan will be working automatically.

6) As the house temperature increases because of cooking, the home-care system turns on the air conditioner.

7) He puts the tableware into the dish washer after he has finished his dinner. And as he goes out the kitchen, the ventilation fan and air conditioner stop working.

Fig. 5 shows the service environment for the home-care services. In Fig. 5, the black spots are sensors which collect context information, the bold-lined arrows indicate the Smith's path, and ① through ⑨ indicate the services according to Smith's situation. These services are as fol-

lows: ① Managing environment; ② Changing environment; ③ Checking foods; ④ Recommending a cook; ⑤ Working the ventilation fan; ⑥ Working the dishwasher; ⑦ Stopping the ventilation fan; ⑧ Working the air conditiooer; ⑨ Stopping the air conditioner.

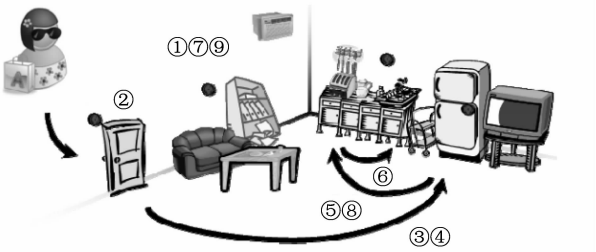

Fig. 5 Service environments for the home-care service scenario

Fig. 6 shows a workflow for the home-care service scenario. It includes the suggested workflow model, such as simple, sub, and parallel workflow model. Let's look at in more details. In Fig. 6, the simple workflow model supports the first service of the home-care services while Mr. Smith has been out. As he returns home, the second service provides him with suitable environments. Then, as he enters the kitchen, the third service is served. Lastly, the fourth service is served, and the service node in the simple workflow needs a sub-workflow to serve more detailed services.

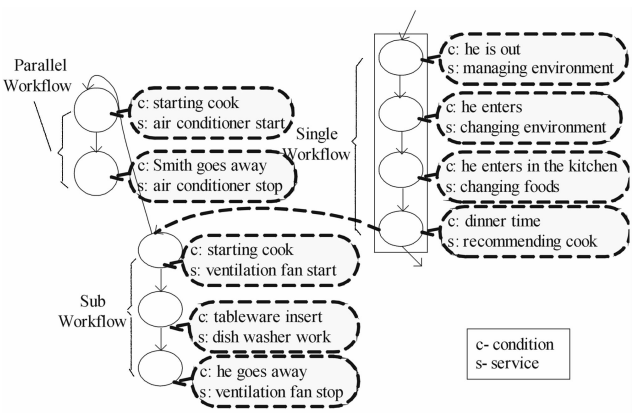

Fig. 6 A workflow for the home-care service scenario

In the fourth node, simple workflow invokes another workflow, a sub workflow. As he starts cooking, the fifth service is served. And he puts the tableware into the dish washer, the sixth service is done. At last, as he gets out of the kitchen, the seventh service is done.

In the fifth service node, the parallel workflow starts. This indicates that the service of "working ventilation fan" is the first node of the sub workflow. As he started cooking, the eighth service, the ventilation fan, is served. Lastly as he gets out the kitchen after his dinner, the ninth service is completed.

Fig. 7 shows the service environment for the office-care services. The second scenario is an office-care. There are two users in this scenario: User A and User B. First, a scenario for User A is as follows:

1) User A usually arrives at his office at 8:30 AM.

2) As he registered his arrival by using a dedicated terminal, the menu for a book return is displayed.

3) As he checks the book return, the office-case system gives information, which is about a location of the book.

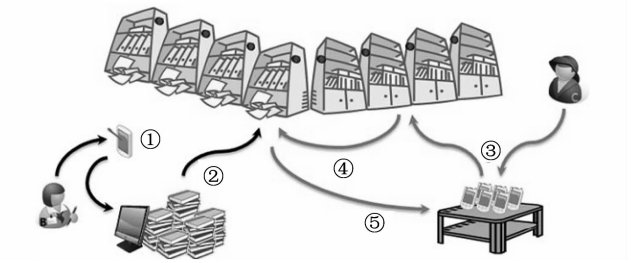

Fig. 7 Office care service environment for multiple workflow services

And a scenario for User B is as follows:

1) User B arrives at the library at 9 o'clock.

2) As his ID card is checked, the location of the reserved books is displayed.

3) If the reserved books are checked out, a message about lending books is displayed.

4) If the non-reserved books are checked, the office-care system displays this message: "This book was already reserved. Would you put a hold on later?"

5) User B checks out the reservations and exits the library.

In Fig. 7, ① through ⑤ indicate the services according to the situation of the User A and User B. These services are as follows: ① Display for a list of books to return; ② Guide for a location of books; ③ Guide for the list of a user's interested books; ④ Book reservation; ⑤ Terminal turn off.

Fig. 8 shows the suggested multiple workflow model.

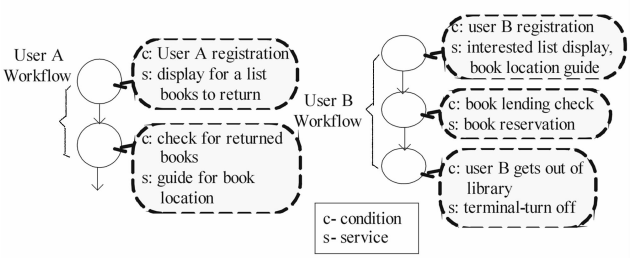

Fig. 8 A workflow for the office-care service scenario

When User A arrived at his office, the first and second services of the office care services were served. When User B entered the library, the office-care system provided him with the third service. Then, if the non-reserved books were checked out, the office-care system provided the fourth service and confirmed another book reservation for User B. As User B exited the library, the system provided him with the fifth service. Therefore, the office-care system supported different services from ① to ⑤ to each user.

5 Conclusion

This paper introduces a context-aware workflow

model to support composite workflows. The suggested model expands the patterns included in the existing context-aware workflows. Because they only support the basic workflow patterns. The suggested model for composite context-aware workflows includes a new context model such as sub and parallel workflow. It is able to use the context information as an execution condition. This condition is used to split one flow into multiple flows or to integrate several flows into one flow. Also the suggested model has the advantage which developers can easily build a new context-aware workflow by reusing the existing workflows. Therefore, they can get high development productivity. It is expected to make it easier to develop applications related to context-aware workflow services on ubiquitous computing environments.

In the future works, we will progress a study to develop a variety of workflow patterns. In order for us to do this, we will design and implement a workflow development environment. This development environment will increase a diversity and adaptability in ubiquitous computing environments.

References

[1] S. Chen, Y. Bu, J. Li, X. Tao, J. Lu, 2006. Toward Context-Awareness: A Workflow Embedded Middleware. Proceedings of UIC'06, LNCS 4159, p. 766-775.

[2] F. Tang, M. Guo, M. Dong, M. Li, H. Guan, 2008. Towards Context-Aware Workflow Management for Ubiquitous Computing. Proceedings of ICESS'08, p. 221-228.

[3] L. Ardissono, A. Di Leva, G. Petrone, M. Segnan, M. Sonnessa, 2005. Adaptive Medical Workflow Management for a Context-dependent Home Healthcare Assistance Service. Proceedings of the CWS'05, p. 59-68.

[4] Dimka Karastoyanova, Alejandro Buchmann, 2004. Extending Web Service Flow Models to Provide for Adaptability. OOPSLA 2004 Workshop on Best Practices and Methodologies in Service-oriented Architectures: Paving the Way to Web-services Success, Vancouver, Canada.

[5] SdfAbdelkarim Erradi, Piyush Maheshwari, Srinivas Padmanabhuni, 2005. Towards a Policy-Driven Framework for Adaptive Web Services Composition, Proceedings of the International Conference on Next Generation Web Services Practices, p. 33-38.

[6] Anand Ranganathan, Scott McFaddin, 2004. Using Workflows to Coordinate Web Services in Pervasive Computing Environments. Proceedings of ICWS'04, p. 189-197.

[7] W. M. P van der Aalst, A. H. M. ter Hofstede, B. Kiepuszewski, A. P. Barros, 2003. Workflow patterns. *Distributed and Parallel Databases*, 14(1): 5-51.

[8] Tony, Andrews, Francisco, Curbera, et al, 2003. Business Process Execution Language for Web Services, BEA Systems. Microsoft Corp. IBM Corp.

[9] Dr. Frank Leymann, 2001. Web Services Flow Language (WSFL 1.0), Distinguished Engineer Member IBM Academy of Technology, IBM Software Group.

[10] Satish, Thatte, 2001. XLANG: Web Services for Business Process Design, Microsoft Corp.

[11] Anders Mller, Michael I. Schwartzbach, 2006. An Introduction to XML AND Web Technologies, Addison-Wesley, ISBN: 0321269667.

[12] Jun Li, Ying-yi Bu, Sha-xun Chen, Xian-ping Tao, Jian Lu, 2006. FollowMe: On Research of Pluggable Infrastructure for Context-Awareness, Proceedings of AINA'06, p. 199-204.

[13] Nick Russell, A. H. M. ter Hofstede, M. P. Wil. van der Aalst, Nataliya Mulyar, 2006. Workflow Control-Flow Patterns: A Revised View, BPM Center Report BPM-06-22, BPMcenter.org.