Operational Semantics of Aspects in Business Process Management

Amin Jalali[1], Petia Wohed[1], and Chun Ouyang[2]

[1] Stockholm University, Sweden
[2] Queensland University of Technology, and NICTA, Australia
{aj,petia}@dsv.su.se, c.ouyang@qut.edu.au

Abstract. Aspect orientation is an important approach to address complexity of cross-cutting concerns in Information Systems. This approach encapsulates these concerns separately and compose them to the main module when needed. Although there are different works which shows how this separation should be performed in process models, the composition of them is an open area. In this paper, we demonstrate the semantics of a service which enables this composition. The result can also be used as a blueprint to implement the service to support aspect orientation in Business Process Management area.

Keywords: Business Process Management, Workflow Management Systems, Aspect Oriented, Coloured Petri Nets, Weaving.

1 Introduction

Reducing complexity in Information Systems is a main concern in both research and industry. One strategy for reducing complexity is separation of concerns. This strategy advocates separating various concerns, like security and privacy, from the main concern. It results in less complex, easily maintainable, and more reusable Information Systems. Separation of concerns is addressed through the Aspect Oriented paradigm. This paradigm has been well researched and implemented in programming, where languages such as AspectJ have been developed. However, the research on aspect orientation for Business Process Management is still at its beginning. While some efforts have been made proposing Aspect Oriented Business Process Modelling (e.g. [3,4,5]), it has not yet been investigated how to enact such process models in a Workflow Management System. We create a Coloured Petri Net (CPN) specification for the semantics of a so-called Aspect Service that extends the capability of a Workflow Management System with support for execution of aspect oriented business process models. The design specification of the Aspect Service has been inspected through state space analysis. In this paper, we briefly present the ideas of the CPN solution.

2 Running Example

For explaining the Aspect Service, we use an abstract example (see Figure 1). The example contains a main process with three aspects X,Y and Z, which are

P. Herrero et al. (Eds.): OTM 2012 Workshops, LNCS 7567, pp. 649–653, 2012.

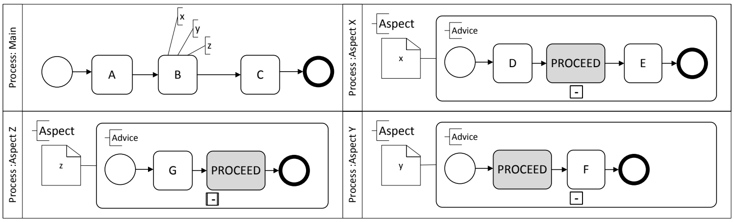

Fig. 1. An abstract example of an aspect oriented process model based on AO4BPMN

defined for one of its activities, activity B. The enactment of the business processes is managed through a WfMS. It results in four different process instances. The Aspect Service takes care of the weaving of the aspects to the main process. This means that the advices are executed in parallel, and a synchronisation towards the main process is made at the `Proceed` placeholder as well as at the end of the execution of the advices. In this particular example, the execution sequence of the activities will be A, followed by D and G in parallel, then B, followed by E and F in parallel, and finally C. This is written as regular expression $A(D||G)B(E||F)C$. In aspect oriented terminology, the activity B is called an advised joint point. An advised joint point contains a so called pointcut condition. If the condition is satisfied the associated advice is triggered. An aspect can contain several advice processes.

3 Overview of the Solution

We define the Aspect Service as a sub-service of the Worklet Service [2]. The Worklet Service is designed to support flexibility in business process management and has certain built-in functions, e.g. suspension of an work item, which are useful for the Aspect Service. The Worklet Specifications repository is used to store the advices. The communication between the Worklet Service and the Workflow engine occurs through a number of message exchanges. These are visualised in the right hand side of Figure 2.

The Aspect Service gets enabled upon receiving a message, which can be one of the following, *ItemPreConstraint*, *ItemPostConstraint* and *CasePostConstraint*. ItemPreConstraint and ItemPostConstraint represent the beginning and ending of a work item while CasePostConstraint represent the ending of a case. When a work item gets enabled, i.e. a constraint of the type ItemPreConstraint is raised by the WfMS, the Aspect Service performs two checks: a check on whether the workitem has a pointcut associated to it and if so, if the pointcut condition is met. If the workitem is not related to a pointcut, or if a pointcut condition is not met, the execution of the workitem is proceed as usual. Otherwise, the Aspect Service starts the weaving of the corresponding aspect(s). The CPN model in Figure 3 captures the behaviour of the service associated with these checks. The weaving is performed in four steps. These are visualized in the left hand side of Figure 2 and described below.

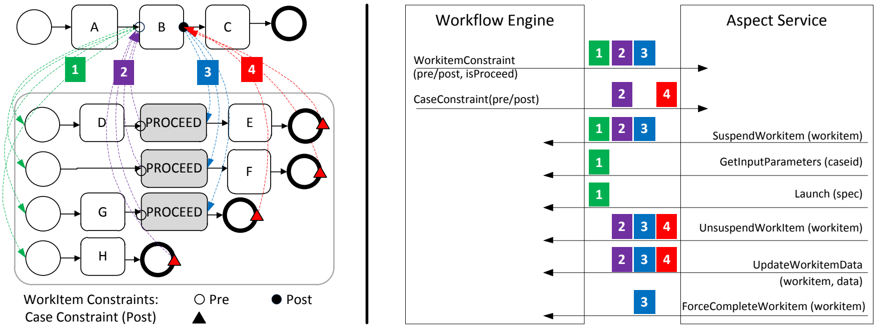

Fig. 2. Weaving steps and corresponding messages

1 *Launching*: When the Aspect Service is activated, it sends a message to the WfMS to suspend the main process. When the suspension is confirmed by the WfMS, the Aspect Service sends messages to the WfMS for launching the relevant advices.

2 *Pausing*: When an advice reaches the `Proceed` placeholder, the Aspect Service sends a message to the WfMS to suspend the advice. However, advices that do not have a `Proceed` placeholder will reach to their End. Once the relevant advices have been suspended or ended, the Aspect Service orders un-suspension of the advised join point.

3 *Resuming*: After the advised join point has been completed, the WfMS raises an ItemPostconstraint. The Aspect Service sends messages for suspension of the advised join point and un-suspension of the corresponding advices. Then, it sends messages to force complete the `Proceed` placeholders, so the advices can be continued.

4 *Finalizing*: When all advices are ended (i.e. their *CasePostconstraints* have been raised), the Aspect Service sends message to the WfMS to unsuspend the advised join point. Hence, the weaving is completed, and the control of the main process is handed back to the WfMS.

During all this steps, the business data is synchronised between the main process and its aspects. In case several advices operate on the same data simultaneously (e.g. activities D and G in Figure 2) the last workitem to complete will overwrite the data stored by the workitems completed earlier.

4 Formal Semantics

The formalisation of the Aspect Service is specified throug a three-level CPN model. The top-level module captures the behaviour of the Initiation of the service, and the second level captures the weaving behaviour (see Figure 3). This model contains four modules capturing steps 1 to 4 described previously. It also contains a module for communicating with the WfMS and performing actions

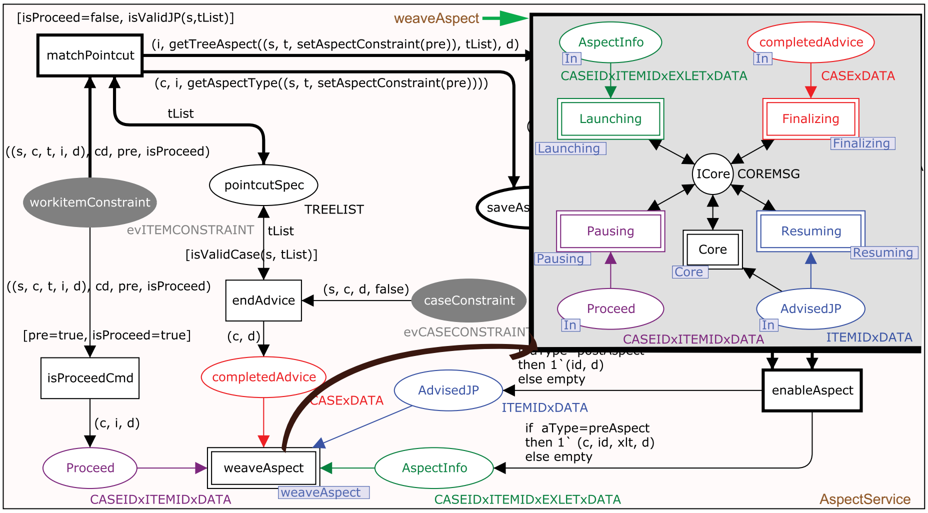

Fig. 3. Tow modules in Aspect Service CPN

for data persistence, which is needed for the weaving of the aspects to the main process. These five modules constitute the third level of the CPN Model.

The model defines 57 colour sets and 33 functions. Full details of the model with definition of the colour sets, variables and functions can be downloaded from [1]. We re-used some of the colour sets, variables and functions from the Worklet Service CPN model [2].

5 Conclusions and Future Work

In this paper, we presented a generic solution for the execution semantics for the weaving of aspects to business processes. The solution is designed based on AO4BPMN and is formalised with CPNs. We have verified the soundness of the design using state space analysis [1].

A direction for future work includes the implementation of the solution in a WfMS. The solution is currently limited to weaving advices in which the Proceed placeholder is only enabled once. This means that the Proceed placeholder can not be included in loops and that in case several Proceed placeholders are defined within the same advice, only one of them should be enabled during the execution of the advice (e.g. as a result of an XOR split). The impact of these limitations, i.e. how frequent such scenarios occur in real life, needs to be further investigated. Moreover, we do not capture the precedence requirement of aspects in this work, which is defined in [5].

References

1. Aspect-Oriented Business Process Management, `http://aobpm.blogs.dsv.su.se/`
2. Adams, M.J.: Facilitating Dynamic Flexibility and Exception Handling for Workflows. PhD thesis, Faculty of IT, QUT (November 2007)
3. Charfi, A.: Aspect-oriented workflow languages: AO4BPEL and applications. PhD thesis, Dr.-Ing. thesis, der Technischen Universitat Darmstadt, Darmstadt (2007)
4. Cappelli, C., et al.: Reflections on the modularity of business process models: The case for introducing the aspect-oriented paradigm. Business Process Management Journal 16, 662–687 (1995)
5. Jalali, A., Wohed, P., Ouyang, C.: Aspect Oriented Business Process Modelling with Precedence. To appear in Proc. 4th Int. Workshop on BPMN (2012)