CSIMQ
Complex
Systems
Informatics
and
Modeling
Quarterly

Weaving of Aspects in Business Process Management

Amin Jalali⋆

Department of Computer and Systems Sciences,
Stockholm University, Stockholm, Sweden

aj@dsv.su.se

Abstract. Separation of cross-cutting concerns is an important issue in business process management, where Aspect-Oriented Business Process Modeling (AO-BPM) aims to support this separation through a new form of encapsulation technique. Although different researchers have investigated how these models can be designed to support separation of non-retroactive cross-cutting concerns, there is no study that defines the separation of retroactive ones. The lack of a unified definition of the syntax and the operational semantics for these models hinders their enactment in practice as well. As a result, the perceived usefulness and usability of these approaches have not yet been investigated so far. Thus, this article fills this gap by formalizing an AO-BPM language and the semantics that can support enactment of such models. The semantics is validated through the state-space analysis technique, and the feasibility of the implementation is also demonstrated. The perceived usefulness and easy to use of the AO-BPM is evaluated by applying the Technology Acceptance Model during a workshop session. The result shows that participants perceived the approach usable and easy to use.

Keywords: Business Process Management, Aspect-Oriented, Weaving, Cross-Cutting Concerns, Modularization.

1 Introduction

Separation of concerns has long been used as an effective approach by people to deal with complex phenomena. This approach helps to reduce the number of dimensions of the complexity by ignoring the concerns which are less relevant and increasing the level of abstraction. Ignoring some aspects, according to the information hiding principle, can enable people to focus on the most relevant aspects of a subject that they are interested in. The application of this approach in information systems is done through decomposing a system into different modules that specify the functionality of the system in overall. Modularizing a system not only increases the level of abstraction but also supports the re-usability of modules, so designing a system can become simpler with the wise choice of modularization techniques.

⋆ Corresponding author

In information systems particular concerns are not limited to one module, and they cross over different modules, so they are known as cross-cutting concerns. It is challenging to encapsulate these concerns in a module since they are scattered through different modules, and those modules are tangled to the concerns. Therefore, it is difficult to hide these concerns when they are not important to be presented to some audiences. Presenting all concerns may affect negatively the understandability of the system by the audience.

There are many cross-cutting concerns in business processes. For instance, concerns related to security and privacy regulations. These sorts of regulations can be imposed in two ways, i.e., i) they can be applied for actions that will happen after the time of establishment (non-retroactively), or ii) they can be applied for actions that also happen before the time of establishment (retroactively). The ex-post-facto law prohibits introducing retroactive regulations in many contexts, e.g. criminal, tax, etc. Thus, in businesses there are many regulations which should only be applied non-retroactively. As changes in regulations can affect many business processes, the systems supporting the affected processes need to be audited and changed accordingly if these regulations are not encapsulated properly.

Aspect Orientation is a paradigm in information systems that aims to encapsulate the cross-cutting concerns. In the Business Process Management (BPM) area, Aspect-Oriented Business Process Modeling (AO-BPM) specifies how these concerns should be modeled when designing business process models. Despite different proposals on how these concerns should be modeled, there is no approach that supports separation of both retroactive and non-retroactive cross-cutting concerns from process models. As a consequence, there is a lack of a general semantics that specifies how these concerns should be enacted.

Therefore, this article extends current AO-BPM approaches by introducing retroactive and non-retroactive cross-cutting concerns. The new modeling language and the semantics of systems that can support enactment of these models are formalized. The result is validated through the state-space analysis technique. In addition, the perceived usefulness and easy to use of the approach is evaluated by applying the Technology Acceptance Model during a workshop session. The result shows that participants perceived the approach usable and easy to use.

The remainder of this article is organized as follows. Section 2 introduces basics of Aspect-Oriented Business Process Management (AO-BPM). Section 3 describes our approach. It specifies the syntax for an AO-BPM language and defines the operational semantics for enacting aspect-oriented business process models. Section 4 validates the models and reports the result of evaluating the Aspect-Oriented Business Process Modeling language. Section 5 discusses the related works. Section 6 concludes the paper and introduces future works.

2 Background

In this section, we explain the basic terms of AO-BPM using a motivating example of a fictious bank.

The left side of Figure 1 illustrates the cross-cutting relations between several regulations and processes in a bank. The vertical and horizontal rectangles show the regulations and core process models respectively. A regulation should be considered in a process model if it is visually crossed over the process models in this figure. For instance, a security regulation is applicable for all processes, while the Anti-Money Laundering (AML) regulation is applicable for only those phases in which the customer can deposit a large amount of money.

The dependency between regulations and processes depends on the law that defines such relations. For instance, consider a new anti-money laundering regulation that enforces the bank to freeze large deposits of customers unless they fill the deposit declaration form. Such regulation can affect many business processes in the bank including the mortgage loan process, LC process, etc., because a large deposit is usually required in these processes. The mortgage loan process
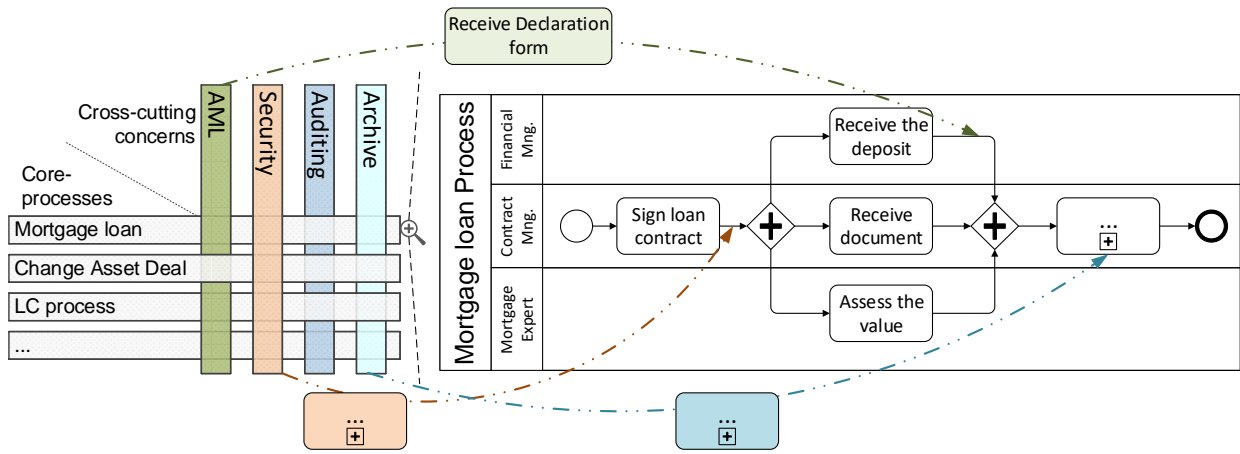
Figure 1. The Motivating Example [1]

requires complying with AML, Security and Archive regulations in our example. Each of these regulations mandates a set of activities to be performed in this process.

The mortgage loan process is shown on the right side of Figure 1. This process starts when a contract manager signs a loan contract with a customer. This contract indicates the regulations that should be considered for each case, which will be signed by both the customer and the bank's employee. Then, the financial manager should receive the deposit from the customer. In parallel, two other activities need to be performed, i.e., the contract manager needs to receive documents from the customer, and mortgage expert needs to assess the mortgage value. The rest of the process is not depicted in details for the sake of simplicity.

Traditionally, all regulations should be captured by related process models, which results in scattering the realization of the regulations in many business process models. For instance, the *receive declaration form* should not only be included in the *mortgage loan process* but also in other applicable processes. The inclusion of this activity (as a realization of the regulation) results in the scattering problem.

Aspect-Oriented Business Process Modeling overcomes this problem by supporting separation of each concern [3], [4], [5]. This approach enables the encapsulation of the regulations and enforces them in the relevant processes when they are needed. Figure 2 shows an abstraction that explains how this approach supports this sort of encapsulation. The vertical and horizontal boxes on the left-hand side of the figure show the cross-cutting concerns and business processes
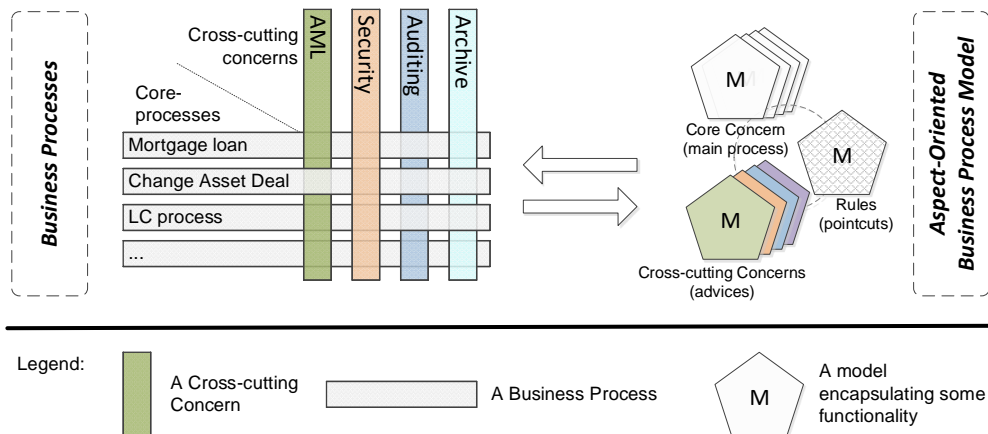

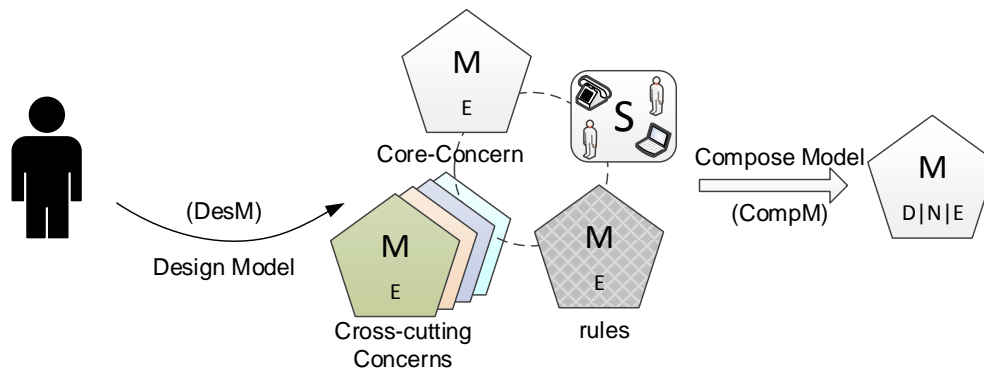**Figure 2.** Aspect-Oriented Business Process Modelling, adopted from [2]

Figure 3. The Static Weaving use case [1]

respectively, which are encapsulated by some models which are shown on the right-hand side of the figure. The functionalities of business processes are encapsulated in models called *core concern*, while the functionalities of cross-cutting concerns are encapsulated in models called *advice*. The relation between these two models is specified by a third model called *pointcuts*. This is very effective modeling since a change in regulation will only affect its model. In addition, if the rule regarding the application of the regulation is changed, the related pointcut will be changed. Thus, it will be much easier to manage both processes and regulations in the bank. The legend that is used in this figure also holds for Figure 4 and Figure 5.

There are two approaches that support the enactment of aspect-oriented business process models, called static and dynamic weaving.

In static weaving [6] (depicted in Figure 3), all fragments of an aspect-oriented business process model that are designed by a process designer are composed together to produce a traditional model. Thus, it does not support the change in cross-cutting regulations or their rules at runtime. Neither does it support the adjustment of business processes to the new regulations at runtime. This technique can be used for introducing regulations that should be applied non-retroactively by producing new versions of a process model for each case, but it is not an efficient way since it causes ongoing configuration costs when enacting business processes. This means that the development team needs to constantly produce a new version of each process model for each case – which is neither desirable nor practical. This technique is a kind of "composite model" use case [7], where a model is composed of several existing models.

In dynamic weaving [8] (depicted in Figure 4), the Business Process Management Systems (BPMSs) should be aware of the semantics of these models [9], and it adapts the process instances at runtime by enforcing the execution of advices based on interpreting pointcuts [10]. Thus, it
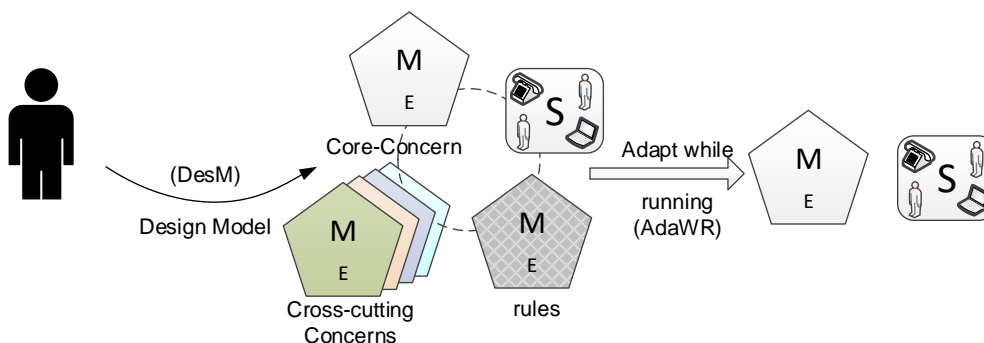

Figure 4. The Dynamic Weaving use case [1]

supports changing cross-cutting regulations or their rules at runtime. A change in regulations will be applied to all cases. This technique is a kind of "Adapt while running" use case [7], where the system knows the semantics of models to execute them correctly.

Although these two approaches can support different use cases, they cannot solve the problem of managing regulations that should be applied non-retroactively in an effective and efficient way. For instance, if we want to introduce the AML regulation in our example, these two approaches are not helpful. The static weaving cannot help since the development team should configure a new process model for each loan application, which is not feasible. Note that there are many regulations in real processes, and the process participants will not know about changes in regulations, so static weaving needs to be performed for every case to guarantee the conformance of processes to regulations. The dynamic weaving cannot help since the new AML regulation will be applied to all cases. Therefore, we need a new sort of weaving that supports enactment of non-retroactive cross-cutting concerns.

3   Approach

In this section we propose a new approach called hybrid weaving, which can support the management of regulations in business processes in an effective and efficient way. In this approach, we divide regulations, which are a sort of cross-cutting concerns, into two categories: non-retroactive and retroactive. Non-retroactive cross-cutting concerns are those concerns which should not be applied to the cases that are created before introducing the concerns. Retroactive cross-cutting concerns are those concerns which should be applied to all cases.

First, we explain our approach by introducing a business process use case for hybrid weaving. Then, we specify the formalization of the operational semantics of the systems that aim to support hybrid weaving to solve the problem.

3.1   The Hybrid Weaving Use Case

Hybrid weaving is a combination of both static and dynamic weaving that can address the gap in regulation management. We introduce this approach as a business process use case by combining several use cases which are introduced by van der Aalst [7], i.e., Design Configurable Model (DesCM), Configure Configurable Model (ConCM), and Adapt While Running (AdaWR) use cases. We also defined a new use case called Compose Configurable Model (CompCM) which is an extension of Compose Model use case. Figure 5 shows the combination of these use cases, which enables hybrid weaving.

In the hybrid weaving use case, the process designer needs to define a configurable aspect-oriented business process model that distinguishes between retroactive and non-retroactive cross-cutting concerns (DesCM). As it can be seen in Figure 5, the pointcut needs to be configurable in these models. Then, the designed models can be composed into a new configurable model (CompCM) by the hybrid weaver. This is a new use case that we introduced, since the static weaving needs to be done on a configurable versions of aspect-oriented business process models so that the result is still configurable. The result has to be configurable since the pointcuts must be configured and be associated with the new model in the next step (ConCM). Finally, the generated configured model will be used by a system which is aware of the semantics of dynamic weaving (AdaWR).

As it can be seen in the Figure 5, the process is complex, and it needs to be formalized in order to investigate its soundness. Therefore, we define the syntax and semantics for these models in next sections. The result can be used by different workflow management systems to extend their functionality. We will further explain these use cases when describing the semantics for aspect-oriented models.
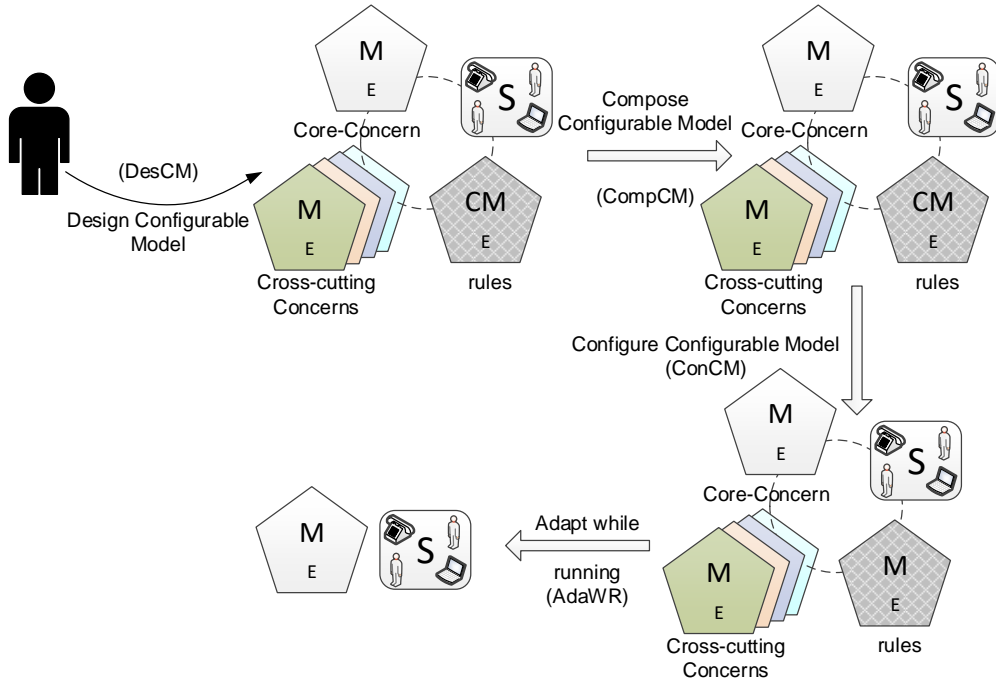
Figure 5. The Hybrid Weaving use case [1]

3.2 Basic Definitions

We define the formal definition and semantics for AO-BPM using Petri-nets. Petri-net is selected because it has a proper semantics, and it can easily be transformed to other business process modeling languages [11] such as Business Process Model and Notation (BPMN) [12], Event-driven Process Chains (EPC) [13], and Yet Another Workflow Language (YAWL) [14].

Definition 1 (P/T-nets [15], [16]). A Place/Transition net (P/T-net) is a tuple $N = (P^N, T^N, F^N)$,[1] where:

- P^N is a finite set of *places*;
- T^N is a finite set of *transitions*, such that $(P^N \cap T^N = \emptyset)$; and
- $F^N \subseteq (P^N \times T^N) \cup (T^N \times P^N)$ is a set of *directed arcs*, called the flow relation.

A place is represented by a circle, and a transition is represented by a rectangle. A flow is represented by a directed arrow connecting a place to a transition, or vice versa. The *preset* of an element $x \in P^N \cup T^N$ in the net N is defined as $\overset{N}{\bullet}x = \{\forall y \in (P^N \cup T^N) | (y, x) \in F^N\}$. The *postset* of an element $x \in P^N \cup T^N$ in the net N is defined as $x \overset{N}{\bullet} = \{\forall y \in (P^N \cup T^N) | (x, y) \in F^N\}$. P, T and F represent the universe of all places, transitions and flows, respectively.

The state of a system is determined by a distribution of tokens in the places of the net. The distribution of tokens in the places of a net is called the *marking* of the net. The transitions of the net can change the markings of the net. A transition is *enabled* iff there is at least one token in every input place. The result of executing a transition is a new marking, where a token is removed from every input place of the transition, and a token is added to every output place of the transition. The initial state of the system is defined by an initial marking. A marking is called *reachable* iff there exists a sequence of enabled transitions whose firing leads from the initial marking to that marking. The sequence of all executed transitions from one marking to another is called the *firing sequence*.

---

[1] The superscript N for a set specifies the net that contains the elements of the set.

A net is *weakly connected*, or simply connected, iff for every two nodes in the net there are a number of flows and nodes that connect one of the nodes to the other. The net is *strongly connected* iff for every two nodes in the net, there are a number of flows and nodes that connect each of them to the other. A marked net is *bounded* iff the set of reachable markings is finite. It is *safe* iff, for any reachable marking, the number of tokens in every place is less than or equal to one. A transition in the net is called *dead* iff there is no reachable markings such that the transition would be enabled. The net is called *live* iff for every reachable marking M' and every transition t, there is another marking M'' which is reachable from M' which enables t.

Definition 2 (Workflow nets [15], [16]). Let $N = (P^N, T^N, F^N)$ be a P/T-net and $\bar{t} \in T \setminus T^N$, where $\bullet\bar{t} = \bar{t}\bullet = \emptyset$. N is a workflow net (WF-net) *iff*: P^N contains an input place i such that $\overset{N}{\bullet}i = \emptyset$; P^N contains an output place o such that $o\overset{N}{\bullet} = \emptyset$; $\bar{N} = (P^N, T^N \cup \{\bar{t}\}, F^N \cup \{(o, \bar{t}), (\bar{t}, i)\}$ is strongly connected.

A workflow net is sound iff the following four properties hold. i) Safeness: the net with the initial given marking is safe; ii) Proper completion: for any marking that is reachable from the initial marking and has a token in the output place, there is no other token in other places; iii) Option to complete: for any marking that is reachable from the initial marking, there is a reachable marking that has a token in the output place (we call this marking the end marking); and iv) Absence of dead tasks: the net contains no dead transitions.

Definition 3 (Core net). A core net is a workflow net that encapsulates the core functionality of a business process. The set of all core nets is denoted by $\mathcal{C}$. Every task in a core net is called a join point.

Definition 4 (Advice). An advice $W = (P^W, T^W \cup \mathbb{P}(\{t^W_{proceed}\}), F^W)$, is a workflow net that encapsulates a cross-cutting functionality, where:

- $W \notin \mathcal{C}$. The advice net is not a part of core nets, and
- $t^W_{proceed}$ is a special task in the advice net which is a place holder for a join point in a core net. $\mathbb{P}$ denotes the power set, so an advice can have zero or one place holders.

An advice W has zero or one $t^W_{proceed}$ place holders. An advice with one place holder is called an around advice. The set of all around advices is denoted by $\mathcal{A}_{\underset{\square}{\bullet}}$. An advice with no place holders is called a free advice. The set of all free advices is denoted by $\mathcal{A}_{\underset{\square}{\circ}}$. The set of all advices is denoted by $\mathcal{A} = \mathcal{A}_{\underset{\square}{\bullet}} \cup \mathcal{A}_{\underset{\square}{\circ}}$, where $\mathcal{A}_{\underset{\square}{\bullet}} \cap \mathcal{A}_{\underset{\square}{\circ}} = \emptyset$.

Definition 5 (Pointcut). A pointcut is a function that relates a join point to a set of advices. There are three sort of pointcuts: before, after and around, denoted by $\mathcal{P}_{\circ\square}$, $\mathcal{P}_{\square\circ}$ and $\mathcal{P}_{\underset{\square}{\bullet}}$, respectively. The set of all pointcuts is denoted by $\mathcal{P} = \mathcal{P}_{\circ\square} \cup \mathcal{P}_{\square\circ} \cup \mathcal{P}_{\underset{\square}{\bullet}}$.

- A before pointcut is a function $\mathcal{P}_{\circ\square} : T \to 2^{\mathcal{A}_{\underset{\square}{\circ}}}$.
- An after pointcut is a function $\mathcal{P}_{\square\circ} : T \to 2^{\mathcal{A}_{\underset{\square}{\circ}}}$.
- An around pointcut is a function $\mathcal{P}_{\underset{\square}{\bullet}} : T \to 2^{\mathcal{A}_{\underset{\square}{\bullet}}}$.

This means that a before and an after pointcuts can relate a set of free advices to a join point, and an around pointcut can relate a set of around advices to a join point. It is also possible that a pointcut relates an empty set of advices to a join point. The join points that are related to at least one advice are called advised join points.

A pointcut can relate a join point to a set of advices retroactively or non-retroactively. The retroactive and non-retroactive pointcuts are denoted by $\mathcal{P}^r$ and $\mathcal{P}^n$ respectively. The annotation applies for all three sorts of pointcuts, e.g. the set of retroactive pointcuts that should be applied before join points can be denoted by $\mathcal{P}^r_{\circ\square}$.

Definition 6 (Aspect-Oriented nets). An Aspect-Oriented net (AO-net) is a tuple $AO = (C, \mathcal{A}, \mathcal{P})$, where: $C \in \mathcal{C}$ is a core net.

A part of the model that will be weaved at design time needs static semantics; while the second part that will be weaved at runtime needs hybrid semantics.

3.3 Static Semantics

The static semantics can be defined by specifying the criteria under which the AO-nets are sound.

Definition 7 (AO-net Soundness). An AO-net $AO = (C, \mathcal{A}, \mathcal{P})$ is sound iff $W = staticWeaving(AO)$ is sound. Algorithm 1 defines $staticWeaving$.

Algorithm 1: Algorithm for weaving an AO-net

---

1 **Algorithm** staticWeaving($AO = (C = (T^C, P^C, F^C), \mathcal{A}, \mathcal{P})$)
2    $W \leftarrow C$;
3   **foreach** *task t in T^W* **do**
4   **if** $\mathcal{P}^n_{\substack{\square \\ \bullet}}(t) \neq \{\}$ **then**
5        $T^W \leftarrow T^W \cup \{t^W_{t.split}\} \cup \{t^W_{t.join}\}$;
6   **else**
7   **if** $\mathcal{P}^n_{\circ\square}(t) \neq \{\}$ **then**
8          $T^W \leftarrow T^W \cup \{t^W_{t.split}\}$;
9   **if** $\mathcal{P}^n_{\square\circ}(t) \neq \{\}$ **then**
10          $T^W \leftarrow T^W \cup \{t^W_{t.join}\}$;
11   **foreach** *Advice A' in $\mathcal{P}^n_{\circ\square}(t)$* **do**
12        $W \leftarrow addAdvice(W, A', t^W_{t.split}, t)$;
13   **foreach** *Advice A' in $\mathcal{P}^n_{\square\circ}(t)$* **do**
14        $W \leftarrow addAdvice(W, A', t, t^W_{t.join})$;
15   **foreach** *Advice A' in $\mathcal{P}^n_{\substack{\square \\ \bullet}}(t)$* **do**
16        $W \leftarrow addAdvice(W, A', t^W_{t.split}, t^W_{t.join})$;
17   **foreach** *Place p in $\bullet t^A_{proceed}$* **do**
18          $F^W \leftarrow F^W \cup \{(p, t)\} \setminus \{(p, t^A_{proceed})\}$;
19   **foreach** *Place p in $t^A_{proceed}\bullet$* **do**
20          $F^W \leftarrow F^W \cup \{(t, p)\} \setminus \{(t^A_{proceed}, p)\}$;
21        $T^W \leftarrow T^W \setminus t^A_{proceed}$;
22   **return** W;
23 **Function** addAdvice($W \in \mathcal{C}, B \in \mathcal{A}, initiator \in T^W, terminator \in T^W$)
1    $A \leftarrow B$;
2    $T^W \leftarrow T^W \cup T^A$;
3    $P^W \leftarrow P^W \cup P^A$;
4    $F^W \leftarrow F^W \cup F^A \cup \{(initiator, i^A)\} \cup \{(o^A, terminator)\}$;
5   **return** W;

This algorithm gets an AO-net and converts it to a Workflow net so that the nets can be analyzed by the techniques available for Workflow Nets. It weaves each related non-retroactive advice to an advised join point in the core net. To discover the advised join points, the algorithm investigates whether every task in the core net is an advised join point. If a task t is an advised join point, the algorithm should weave i) a "before advice" before the task; ii) an "around advice" before and after

the task, and iii) an "after advice" after the task. Therefore, it adds a helper split and joins tasks for weaving advices. Thereafter, it weaves before, after and around advices to the net. To weave an "around advice," the algorithm should also replace the "proceed" to the advised join point. Finally, the algorithm returns the woven net. The function $addAdvice$ relates advice to an advised join point, which is used in weaving before, after and around advices.

The complete semantics of AO-nets needs to be defined dynamically, due to the existence of retroactive advice.

3.4 Hybrid Semantics

This section completes the semantics of Aspect-Oriented Business Process Models by defining hybrid semantics. We used Coloured Petri-Nets (CPN) [17] to formalize the hybrid semantics, because it is a widely-used formalization technique in system design and verification. This technique is also supported by a tool called CPN Tools that enables verification of the result [18]. This technique is also widely used in the domain of workflow management systems, e.g. Adams specifies the semantics of worklet service using Coloured Petri-nets [19]; or Jalali et al. define the dynamic weaving using Coloured Petri-nets [9] which is later implemented and applied in a banking case [8], [10].
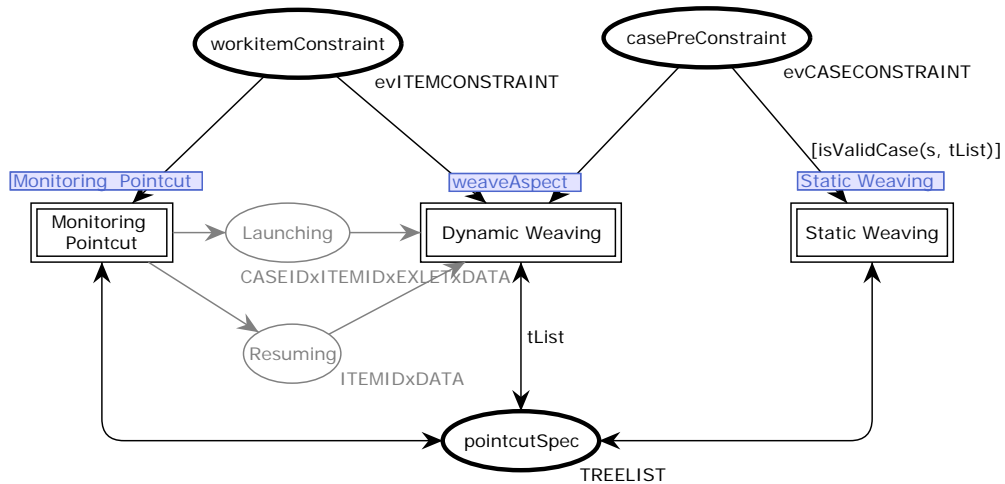

Figure 6. The Hybrid Weaving net [1]

Figure 6 shows the top net of the CPN model for the hybrid semantics. This semantics is defined based on the approach that we defined earlier for hybrid weaving. It contains three subnets, called monitoring pointcuts, dynamic weaving and static weaving:

- Monitoring pointcut net is responsible for monitoring business process instances. A strong monitoring service needs to capture different business process perspectives [20].
- The dynamic weaving net is responsible for enforcing cross-cutting concerns at runtime. The dynamic weaving and monitoring nets define how the Adapt While Running (AdaWR) use case is defined in the hybrid weaving use case. The semantics is defined by Jalali et al. [9].
- The static weaving net is responsible for performing the Compose Configurable Model (CompCM) and Configure Configurable Model (ConCM) use cases.

The monitoring pointcuts and dynamic weaving subnets are defined in [10], where we defined the semantics of dynamic weaving. Thus, we do not elaborate on them in detail here.

The service starts when a case is created in the workflow management systems. For every case, the service will investigate if there is non-retroactive cross-cutting concern that should be weaved into the model. The detailed method which is defined as a model is explained below (see also Figure 7).

Figure 7. The Configuration steps in Hybrid Weaving [1]

Compose Configurable Model:

In this sub- use case, we need to compose a new model that composes non-retroactive advice with the core process specification in a new model.

To perform such an operation, the system reacts for each instance of a business process model that is created as a case in the system. The creation of each case is recognized as an event represented as a token appears in the *casePreConstraint* place. The process is depicted in Figure 7.

The service starts when an instance of a business process model is created in the system, known as a case. This service reads the pointcuts models and investigates if any non-retroactive advice is defined for this case. If none non-retroactive advice is related, then the service will not change the configurable model, and it can be enacted as a model with the help of dynamic weaving semantics, specified by Jalali et al. [9].

However, if a non-retroactive advice is defined for such a case, the service needs to retrieve the specification of process model from the server by sending a request. Note that the request is sent as a message through a channel called *ICore* in our model, and the response will be given from the same channel. Next, the system can cancel the initiated case. The cancellation cannot be performed in the first place, since it might happen before retrieving the case specification due to concurrent nature of message exchange with workflow management systems.

Then, the service needs to retrieve the non-retroactive advice, and it should perform static weaving [6]. The result is a woven model as a new specification. The system needs to upload the new specification, which is performed by the *uploadSpec* transition.

Configure Configurable Model:

At this point, the retroactive advice is related to the previous process model. Thus, the system needs to configure the pointcut model to be related to the newly generated model. Thus, the system gets the *ID* of uploaded specification using the *retrieveSpecID* transition. It will update the pointcut repository using the *updatePointcuts* transition.

Adapt While Running:

At this point, all configuration is applied appropriately. Thus, the system will create a new instance of the new process model by *LaunchNewCase* transition. Note that the creation of such a case will raise a new case constraint event by the workflow management system. This event will make the service active again. However, it will skip the static weaving part since there is no non-retroactive advice related to the new case. Hence, the new case will only go through the dynamic weaving service [9], which handles the Adapt While Running sub- use case.

The result can enable management of both non-retroactive and retroactive cross-cutting concerns in an effective manner.

4 Evaluation

We evaluate three important aspects of our artifacts, i.e. feasibility of the implementation, semantics of the model, and the usefulness and easy to use of aspect-oriented business process modeling.

4.1 Investigating Feasibility of the Implementation

The feasibility of the implementation of hybrid weaving can be evaluated by investigating the feasibility of implementation of its parts, i.e., static and dynamic weaving.

Static Weaving

To investigate the feasibility of the implementation of static weaving, we have implemented a prototype tool based on the static weaving algorithm to investigate the soundness of different scenarios of aspect-oriented business process models. The static weaving algorithm is implemented using the jBPT library [21]. This library makes it easier to extend the implementation to support the weaving of other process model notations like EPC and BPMN. The algorithm gets a core-net and set of advices as Petri-nets in the Petri Net Markup Language (PNML) format. It also gets the pointcuts in the XML format which is defined in [8]. The result is a woven net as Petri-nets in the PNML format. This format is supported by many Petri-net editors. To analyze the result, WoPeD [22] is used. WoPeD supports analysis of Petri-nets using Woflan - an advanced Petri-net-based workflow analyser [23].

The result of analyzing different scenarios shows that the woven result is always sound only if i) there is no advised join points in the model for which more than one advice is defined, or ii) no any around advice is defined for an advised join point for which more than one advice is defined. Otherwise, the soundness must be checked by the tool.

Figure 8 shows three nets for a scenario that can have deadlock after weaving. It contains a core process containing three tasks, i.e. A, B and C. A pointcut relates two advices to task B in the main process model. One of the advices is "BEFORE" advice (named before), and the other one is "AROUND" (named around_loop). As it can be seen in the around advice, it is possible to fire the PROCEED by firing the loop transition. Figure 9 shows the result of the weaving of the nets for this scenario. The graph bellow the Petri-net shows the reachability graph, which is generated by WoPeD. The reachability graph shows all possible reachable states in the process model. The analysis based on the reachability graph is a very powerful method because it can be used to prove or disprove different kinds of properties [24].

The reachability graph shows that there is a possible deadlock in this process model. The deadlock can occur if the loop transition is fired. The problem is that the advised join point (task B) requires a token in both input places. However, there will be no token in the output place of the "before advice" net because it is consumed by the advised join point previously. This introduces a deadlock in the woven net. The static weaving can help us to analyze the soundness of aspect-oriented business process models.
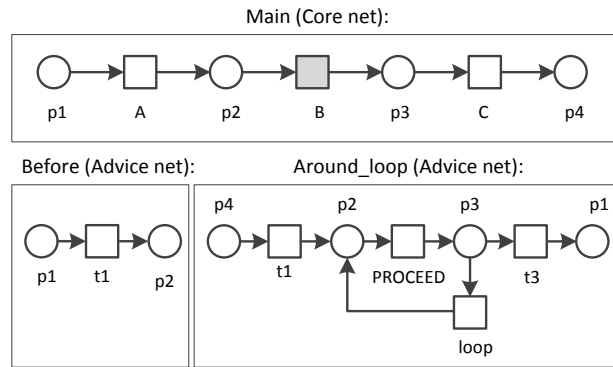
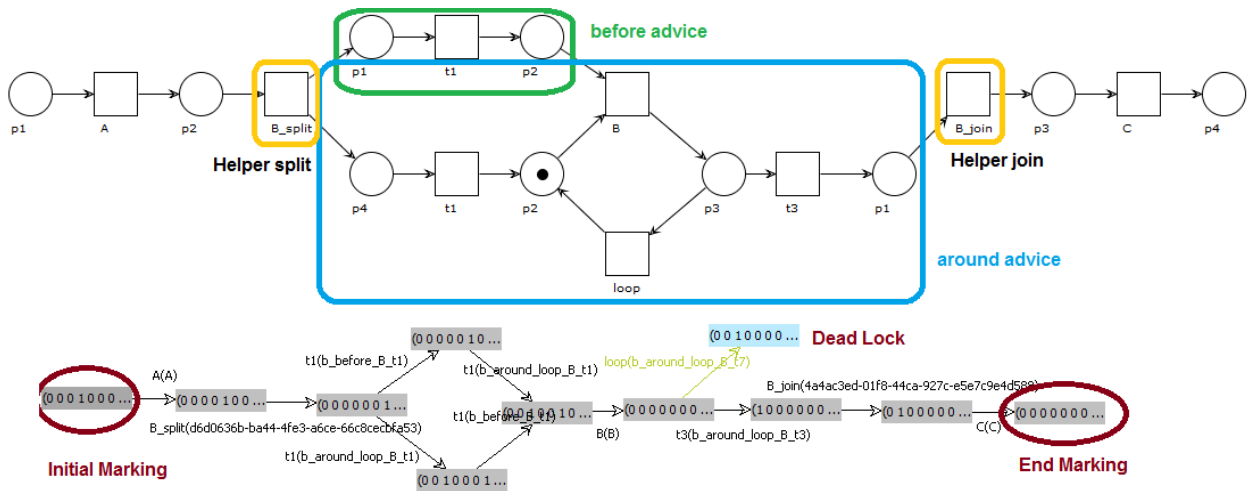**Figure 8.** Aspect-Oriented nets defined for our sample scenario [6]

Figure 9. Weaving the sample scenario [6]

Dynamic Weaving

The feasibility of implementation of dynamic weaving was investigated by us in another work [10]. The implementation is an instance of monitoring pointcuts and dynamic weaving sub-nets in our semantics. Therefore, the whole model can be used as a blueprint to support enactment of aspect-oriented business process models in Business Process Management Systems.

4.2   Verification of Semantics of the Model

The presented CPN model allows us to verify the design of our artifact using state space analysis technique. The verification is a complicated task due to the numerous types of advices and their possible combinations that may be triggered by an activity. We divide all the possible situations into three groups and define three abstract scenarios to characterize them. We aimed to support both retroactive and non-retroactive advices using our approach. Therefore, the following scenarios are defined to investigate the correctness of our approach:

- Scenario 1: Introducing non-retroactive advice for a business process model while some of its instances are currently executing.
- Scenario 2: Introducing retroactive advice for a business process model while some of its instances are currently executing.
- Scenario 3: Introducing both a retroactive and non-retroactive advice for a business process model while some of its instances are currently executing.

35

We applied the state space analysis technique to investigate if:

- In Scenario 1, the non-retroactive advice is only applied to new cases, and old cases are not affected.
- In Scenario 2, the retroactive advice is applied to both new and old cases.
- In Scenario 3, the non-retroactive advice is only applied to new cases, and old cases are not affected; while the retroactive advice is applied for both new and old cases.

The result of applying state-space analysis techniques for these scenarios shows that all these requirements are held by the model. In addition, the number of nodes and arcs for both State Space results and Strongly Connected Component (SCC) graphs are the same, which indicates the lack of cycle in our model. In each scenario, there is only one dead marking, which is equivalent to the home marking. This indicates that there is only one final state in this model from which no other states can be reached, which is the end marking. In addition, the reports state that there are no infinite occurrence sequences or any live transition instances. The result of the general state space analysis shows that our approach does not have any deadlocks or live-locks, and the net will be properly ended.

4.3 Evaluating Usefulness and Easy to Use Features

To evaluate whether aspect-oriented business process models are useful and easy to use, we applied the evaluation method which is proposed by Moody [25]. This method is proposed to evaluate the information systems' artifacts based on the technology acceptance model [26]. This method considers the *actual usage* of an artifact as a variable which depends on the *intention to use* of the artifact by users.

The perceptions of the users influence the intention to use. This perception can be evaluated for *efficiency* and *effectiveness* of the artifacts, which is called *perceived ease of use* and *perceived usefulness* respectively. The effort to complete a task is defined as efficiency, while the scale of improving the quality of the result is defined as the effectiveness. This method is widely used to evaluate artifacts in the business process management area (e.g. [27], [28]).

This is the first attempt to evaluate perceived ease of use and usefulness of an AO-BPM approach, to the best of our knowledge. In this attempt, we evaluated two approaches, i.e. the imperative (the approach that is presented in this article) and hybrid aspect-oriented business process modeling approaches. The result of evaluation for the hybrid approach is published in [1]. The evaluation is performed through a workshop with 28 students, of which 20 were bachelor and 8 were master level students. These students have been studying a course on business process management. They learnt about business process modeling, analysis, configuration, implementation, monitoring and mining. They experienced many tools during the course like WoPeD for designing and analyzing business process models using Petri-nets; Signavio for designing BPMN models; YAWL for modeling, analysis and enacting business processes using YAWL notation; ProM for performing process mining activities. Thus, they had quite a good background in this area, and they can be considered as good candidates as novice process analysts. The age of students varies, and 11 participants were females, and 17 were males.

We experienced better outcomes when students performed process design activities in groups of two or three. Thus, in the workshop, we divided students into ten groups randomly, i.e. eight groups of three and two groups of two students. We gave each group descriptions of dealing business processes of a bank that we have reported in [10]. We asked students to design process models for the given cases. Then, we asked them to apply a change in the model. Later, we introduced the aspect-oriented business process modeling, and we asked them to design the process using this technique. Afterwards, we asked them to apply the change in the new model. Finally, we asked every participant to fill a survey individually. The survey was designed based on the questionnaire

given in [26]. This questionnaire was chosen with the aim of measuring the perceived usefulness and easy to use of our approach based on the framework defined in this article. The details about questions can be found in Appendix.

The survey contains two sets of questions to measure if our proposed approach improves usefulness (first set) and ease of use (second set) with respect to a standard modeling approach. Students could choose answers based on scales from *extremely likely* to *extremely unlikely*, which are mapped to numbers from 7 to 1 respectively [26]. Figure 10 shows how our approach is perceived regarding usefulness and ease of use. The medians for both measures are above five, which shows that the majority of participants found the approach useful and easy to use. In general, the usefulness is perceived slightly better than ease of use by the participants.
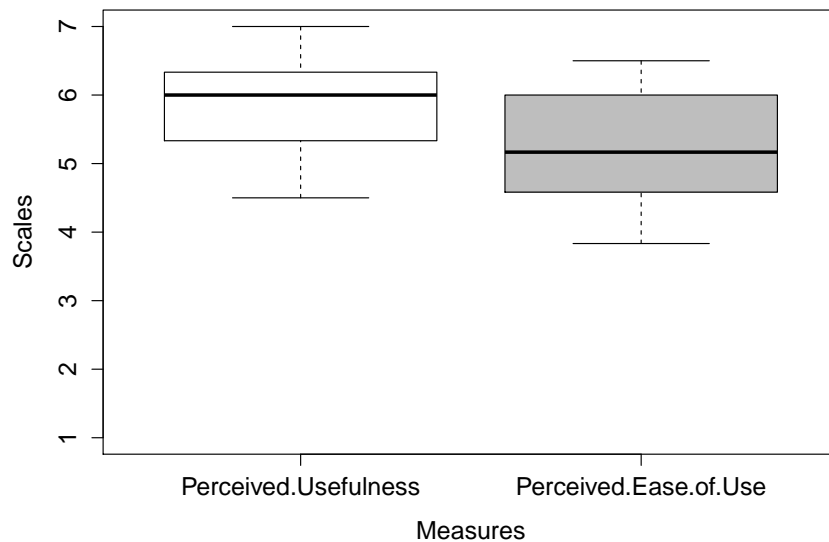

Figure 10. Perceived Usefulness and Ease of Use of our approach in General

There were some comments by the audience about their overall opinion of applying AO-BPM and its drawbacks. We received the following comments regarding the following question:
What is your overall opinion of applying AO-BPM?

- "It is useful and gives other ways on how to think to solve problems or challenges."
- "Very flexible."
- "It enhances the flexibility of the model; easier to change/modify."
- "It was an effective way to show and apply changes."
- "It makes modeling complex situations easier and more understandable."
- "More accurate models."

We also received the following comments regarding the following question:
What are the drawbacks of using AO-BPM in modeling business processes from your viewpoint?

- "Maybe it needs more practice and needs to be discussed with others to get more insights."
- "Harder to get the full picture of the process."
- "Could be hard to get a new overall view of the process model."
- "It is harder to get a quick view of all the aspects of the model."
- "It increases the number of models for one process."

It is correct that the overall picture of the process would be harder to grasp by looking to the aspect-oriented business process models since one should understand the whole process by looking

at its fragments. However, the static weaving is a solution to overcome this problem since it can merge all pieces and generate a flattened version of the process, from which it is easier to get the overall picture. Based on our evaluation, we hope that the proposed modeling approach can provide an additional instrument to support the design of complex process models.

It should be noted that we only evaluated the easy to understand and easy to use of our approach from students point of view. This means that we cannot conclude if our approach can be accepted by practitioners. Such a conclusion requires further evaluation of the approach in future.

5   Related Work

In this section, we discuss related works. The separation of cross-cutting concerns has been investigated in different areas. This investigation was performed first in the Programming area, called Aspect Oriented Programming (AOP) [29], [30]. This new encapsulation technique helped programmers to separate different concerns like security and privacy, which helped them to overcome the complexity of developing very complex programs.

This idea had inspired two other paradigms, i.e., requirement engineering [31] and service composition [32], [33], [34], [35]. In requirement engineering, aspect-oriented approach aims to support separation of cross-cutting functional and non-functional properties of requirements to enhance their mapping and development [31]. In service composition, it aims at enabling developers to encapsulate cross-cutting concerns as modules and enforcing their execution at the right moment. Aspect-Oriented Service Composition inspired researchers to work on separation of concerns in the context of business process models. In business process management, there are different attempts to support separation of cross-cutting concerns from process models at different phases.

To support the design phase, the idea of aspect-oriented business process modeling is proposed by Charfi et al. [5] and Cappelli et al. [3]. Cappelli et al. [3] developed a tool to support the building of such models, called CrossOryX, which is based on Oryx [36] framework. The definitions of aspect-oriented business process modeling are formalized by Jalali et al. [4]. These works focus on BPMN notation, which is an imperative business process modeling. Thereafter, Jalali et al. proposed two different approaches for declarative [37] and hybrid [38] aspect-oriented business process modeling.

In addition to these works, some researchers proposed some approaches to support the discovery of aspect-oriented business process models, e.g. [39], [40], [41], [42]. The list of some open issues in aspect-oriented business process modeling is provided by Santos et al. [43]. Jalali proposed a framework that enables comparison of these approaches [44].

To support the analysis phase, there has been a gap in the definition of analysis technique to investigate the soundness of aspect-oriented business process models. This gap is addressed by this article by formalizing the definition of aspect-oriented business process modeling using the Petri-nets notation and definition of the static weaving algorithm. The implementation of this algorithm enabled us to analyse aspect-oriented business process models using existing analysis technique such as place invariants [45], transition invariants [46], reachability graph [47], woflan [48], etc.

To support the configuration/implementation phase, Jalali et al. [9] defined the operational semantics of dynamic weaving using Coloured Petri-nets. This approach only supports the definition and management of the retroactive cross-cutting concern. We extended this definition to support non-retroactive approaches in this article.

To support the run/adjustment phase, Jalali et al. implemented service in Yet Another Workflow Language (YAWL) [14], called Aspect Service [8], [10]. They also explained how this approach could support agile business process development in organizations [49].

To support the data-based analysis phase, Jalali proposed an approach to do aspect mining in BPM [50]. The discovery part of the approach is further investigated by Brandão et al. to identify cross-cutting concerns automatically from event logs using cloning and clustering methods [51].

All of these approaches only focus on managing retroactive cross-cutting concerns, and they cannot handle non-retroactive ones. This work extends the area of Aspect-Oriented Business Process Management by defining a new approach to fill this gap.

6 Conclusion

In this article, we defined an Aspect-Oriented Business Process Modeling Language that can support separation of both non-retroactive and retroactive cross-cutting concerns in business process management. The syntax and the semantics of the language are formalized. The formalization of syntax is defined based on Petri-nets. The semantics are defined by formalizing both static and hybrid semantics. The static semantics are specified through the definition of the static weaving algorithm, and the hybrid semantics are specified using Coloured Petri-Nets.

The formalized syntax and semantics not only enables development of tools to support this notation, but it also enables checking the correctness of these models. Such checking is important since it enables analysis of business process models; and it enables designers to find and fix the design problems.

The approach is evaluated based on three important aspects, i.e. *feasibility study of the implementation*, *verification of the semantics* of the model, and *evaluation of the usefulness and easy to use* of aspect-oriented business process modeling.

The *feasibility study of the implementation* is demonstrated by implementing both static weaving and dynamic weaving in a business process management system. The implementations show the feasibility of implementing the defined syntax and semantics in current business process management systems, which can help to extend the functionality of other workflow management systems or business process management systems based on these definitions.

The *verification of the semantics* is performed by implementing the semantics in CPN Tools and applying state space analysis technique. Such verification ensures the lack of design problems in defined syntax and semantics, which is important and hard to check due to many possible concurrent message passing that can be performed in the business process management systems.

The *evaluation of usefulness and easy to use* of the language is performed by applying the Technology Acceptance Model during a workshop session. This evaluation shows that users perceived the approach both easy to understand and easy to use. This is a very important evaluation aspect since the artifacts that are hard to be used by users may be not used in practice.

The result can help different vendors and researchers to extend the capability of Business Process Management Systems to support separation of cross-cutting concerns in future.

References

[1] A. Jalali, F. M. Maggi, and H. A. Reijers, "A hybrid approach for aspect-oriented business process modeling," *Journal of Software: Evolution and Process*, 2018. [Online]. Available: https://doi.org/10.1002/smr.1931

[2] A. Jalali, "Aspect-Oriented Business Process Management," Ph.D. dissertation, Department of Computer and Systems Sciences, Stockholm University, December 2016.

[3] C. Cappelli, F. Santoro, J. do Prado Leite, T. Batista, A. Medeiros, and C. Romeiro, "Reflections on the modularity of business process models: The case for introducing the aspect-oriented paradigm," *BPM Journal*, vol. 16, pp. 662–687, 2010. [Online]. Available: https://doi.org/10.1108/14637151011065955

[4] A. Jalali, P. Wohed, and C. Ouyang, "Aspect oriented business process modelling with precedence," in *Business Process Model and Notation*. Lecture Notes in Business Information Processing, Springer, vol. 125, pp. 23–37, 2012. [Online]. Available: https://doi.org/10.1007/978-3-642-33155-8_3

[5] A. Charfi, H. Müller, and M. Mezini, "Aspect-oriented business process modeling with AO4BPMN," in *Modelling Foundations and Applications*. Lecture Notes in Computer Science, Springer, vol. 6138, pp. 48–61, 2010. [Online]. Available: https://doi.org/10.1007/978-3-642-13595-8_6

[6] A. Jalali, "Static weaving in aspect oriented business process management," in *Conceptual Modeling*. Lecture Notes in Computer Science, Springer, vol. 9381, pp. 548–557, 2015. [Online]. Available: https://doi.org/10.1007/978-3-319-25264-3_41

[7] W. M. P. van der Aalst, "Business process management: A comprehensive survey," *ISRN Software Engineering*, vol. 2013, p. 37, 2013. [Online]. Available: https://doi.org/10.1155/2013/507984

[8] A. Jalali, P. Wohed, C. Ouyang, and P. Johannesson, "Dynamic weaving in aspect oriented business process management," in *On the Move to Meaningful Internet Systems: OTM 2013 Conferences*. Lecture Notes in Computer Science, Springer, vol. 8185, pp. 2–20, 2013. [Online]. Available: https://doi.org/10.1007/978-3-642-41030-7_2

[9] A. Jalali, P. Wohed, and C. Ouyang, "Operational semantics of aspects in business process management," in *On the Move to Meaningful Internet Systems: OTM 2012 Workshops*. Lecture Notes in Computer Science, Springer, vol. 7567, pp. 649–653, 2012. [Online]. Available: https://doi.org/10.1007/978-3-642-33618-8_85

[10] A. Jalali, C. Ouyang, P. Wohed, and P. Johannesson, "Supporting aspect orientation in business process management," *Software & Systems Modeling*, vol. 16, no. 3, pp. 903–925, 2017. [Online]. Available: https://doi.org/10.1007/s10270-015-0496-7

[11] N. Lohmann, E. Verbeek, and R. Dijkman, "Petri net transformations for business processes – a survey," in *Transactions on Petri Nets and Other Models of Concurrency II*. Lecture Notes in Computer Science, Springer, vol. 5460, pp. 46–63, 2009. [Online]. Available: https://doi.org/10.1007/978-3-642-00899-3_3

[12] I. O. Object Management Group, "Business Process Model and Notation (BPMN)," Object Management Group, Inc. (OMG), Tech. Rep., 2013. [Online]. Available: http://www.omg.org/spec/BPMN/2.0.1/PDF

[13] G. Keller, A. Scheer, and M. Nüttgens, *Semantische Prozeßmodellierung auf der Grundlage" Ereignisgesteuerter Prozeßketten (EPK)"*. Inst. für Wirtschaftsinformatik, 1992.

[14] W. M. P. van der Aalst, L. Aldred, M. Dumas, and A. H. M. Hofstede, "Design and implementation of the YAWL system," in *Advanced Information Systems Engineering*. Lecture Notes in Computer Science, Springer, vol. 3084, pp. 142–159, 2004. [Online]. Available: https://doi.org/10.1007/978-3-540-25975-6_12

[15] W. M. P. van der Aalst, "Verification of workflow nets," in *Application and Theory of Petri Nets*. Lecture Notes in Computer Science, Springer, vol. 1248, pp. 407–426, 1997. [Online]. Available: https://doi.org/10.1007/3-540-63139-9_48

[16] W. M. P. van der Aalst, T. Weijters, and L. Maruster, "Workflow mining: discovering process models from event logs," *Knowledge and Data Engineering*, vol. 16, no. 9, pp. 1128–1142, 2004. [Online]. Available: https://doi.org/10.1109/TKDE.2004.47

[17] K. Jensen, "Coloured petri nets," in *Petri nets: central models and their properties*. Lecture Notes in Computer Science, Springer, vol. 254, pp. 248–299, 1987. [Online]. Available: https://doi.org/10.1007/978-3-540-47919-2_10

[18] K. Jensen, L. M. Kristensen, and L. Wells, "Coloured petri nets and cpn tools for modelling and validation of concurrent systems," *International Journal on Software Tools for Technology Transfer*, vol. 9, no. 3-4, pp. 213–254, 2007. [Online]. Available: https://doi.org/10.1007/s10009-007-0038-x

[19] M. J. Adams, "Facilitating dynamic flexibility and exception handling for workflows," Ph.D. dissertation, Queensland University of Technology Brisbane, Australia 14, 2007.

[20] A. Jalali and P. Johannesson, "Multi-perspective business process monitoring," in *Enterprise, Business-Process and Information Systems Modeling*. LNBIP, Springer, vol. 147, pp. 199–213, 2013. [Online]. Available: https://doi.org/10.1007/978-3-642-38484-4_15

[21] A. Polyvyanyy and M. Weidlich, "Towards a compendium of process technologies: The jBPT library for process model analysis," in *CAiSE'13 Forum*, pp. 106–113, 2013.

[22] A. Eckleder and T. Freytag, "Woped a tool for teaching, analyzing and visualizing workflow nets," *Petri Net Newsletter*, vol. 75, pp. 3–8, 2008.

[23] W. M. P. van der Aalst, "Woflan: a petri-net-based workflow analyzer," *Systems Analysis Modelling Simulation*, vol. 35, no. 3, pp. 345–358, 1999.

[24] W. M. P. van der Aalst, "Interval timed coloured petri nets and their analysis," in *Application and Theory of Petri Nets*. Lecture Notes in Computer Science, Springer, vol. 691, pp. 453–472, 1993. [Online]. Available: https://doi.org/10.1007/3-540-56863-8_61

[25] D. L. Moody, "The method evaluation model: a theoretical model for validating information systems design methods," *ECIS 2003 proceedings*, p. 79, 2003.

[26] F. Davis, "Perceived usefulness, perceived ease of use, and user acceptance of information technology," *MIS Quarterly*, vol. 13, no. 3, pp. pp. 319–340, 1989. [Online]. Available: https://doi.org/10.2307/249008

[27] M. La Rosa, P. Wohed, J. Mendling, A. ter Hofstede, H. Reijers, and W. M. P. van der Aalst, "Managing Process Model Complexity Via Abstract Syntax Modifications," *IEEE Trans. Industrial Informatics*, vol. 7, no. 4, pp. 614–629, 2011. [Online]. Available: https://doi.org/10.1109/TII.2011.2166795

[28] M. La Rosa, A. ter Hofstede, P. Wohed, H. Reijers, J. Mendling, and W. M. P. van der Aalst, "Managing process model complexity via concrete syntax modifications," *Industrial Informatics, IEEE Transactions on*, vol. 7, no. 2, pp. 255–265, May 2011. [Online]. Available: https://doi.org/10.1109/tii.2011.2124467

[29] R. Filman, D. Friedman, and P. Norvig, "Aspect-oriented programming is quantification and obliviousness," 2000.

[30] T. Elrad, R. Filman, and A. Bader, "Aspect-oriented programming: Introduction," *Commun. ACM*, vol. 44, no. 10, pp. 29–32, 2001. [Online]. Available: https://doi.org/10.1145/383845.383853

[31] A. Rashid, P. Sawyer, A. Moreira, and J. Araújo, "Early aspects: A model for aspect-oriented requirements engineering," in *Proceedings IEEE Joint International Conference on Requirements Engineering*, pp. 199–202, 2002. [Online]. Available: https://doi.org/10.1109/ICRE.2002.1048526

[32] A. Charfi and M. Mezini, "Aspect-oriented web service composition with AO4BPEL," in *Web Services*. Lecture Notes in Computer Science, Springer, vol. 3250, pp. 168–182, 2004. [Online]. Available: https://doi.org/10.1007/978-3-540-30209-4_13

[33] A. Charfi and M. Mezini, "Aspect-oriented workflow languages," in *On the Move to Meaningful Internet Systems 2006: CoopIS, DOA, GADA, and ODBASE*. Lecture Notes in Computer Science, Springer, vol. 4275, pp. 183–200, 2006. [Online]. Available: https://doi.org/10.1007/11914853_12

[34] A. Charfi and M. Mezini, "AO4BPEL: An aspect-oriented extension to BPEL," *World Wide Web*, vol. 10, no. 3, pp. 309–344, 2007. [Online]. Available: https://doi.org/10.1007/s11280-006-0016-3

[35] A. Charfi, "Aspect-oriented workflow languages: AO4BPEL and applications," Ph.D. dissertation, der Technischen Universitat Darmstadt, Darmstadt, November 2007. [Online]. Available: http://tuprints.ulb.tu-darmstadt.de/852/1/thesischarfifinal.pdf

[36] G. Decker, H. Overdick, and M. Weske, "Oryx - an open modeling platform for the bpm community," in *Business Process Management.* Lecture Notes in Computer Science, Springer, vol. 5240, pp. 382–385, 2008. [Online]. Available: https://doi.org/10.1007/978-3-540-85758-7_29

[37] A. Jalali and I. Bider, "Towards aspect oriented adaptive case management," in *Enterprise Distributed Object Computing Conference Workshops and Demonstrations (EDOCW), 2014 IEEE 18th International*, pp. 143–151, 2014. [Online]. Available: https://doi.org/10.1109/EDOCW.2014.30

[38] A. Jalali, F. M. Maggi, and H. A. Reijers, "Enhancing aspect-oriented business process modeling with declarative rules," in *Conceptual Modeling. ER 2015.* Lecture Notes in Computer Science, Springer, vol. 9381, pp. 108–115, 2015. [Online]. Available: https://doi.org/10.1007/978-3-319-25264-3_8

[39] C. Di Francescomarino and P. Tonella, "Crosscutting concern documentation by visual query of business processes," in *Business Process Management Workshops.* Lecture Notes in Business Information Processing, Springer, vol. 17, pp. 18–31, 2009. [Online]. Available: https://doi.org/10.1007/978-3-642-00328-8_3

[40] C. Di Francescomarino and P. Tonella, "Crosscutting concern mining in business processes," *Software, IET*, vol. 5, no. 6, pp. 552–562, 2011. [Online]. Available: https://doi.org/10.1049/iet-sen.2010.0145

[41] C. Ghidini, C. Francescomarino, M. Rospocher, P. Tonella, and L. Serafini, "Semantics-based aspect-oriented management of exceptional flows in business processes," *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, vol. 42, no. 1, pp. 25–37, 2012. [Online]. Available: https://doi.org/10.1109/TSMCC.2011.2138133

[42] F. Santos, C. Cappelli, F. Santoro, J. do Prado Leite, and T. Batista, "Analysis of heuristics to identify crosscutting concerns in business process models," in *Proceedings of the 27th Annual ACM Symposium on Applied Computing.* ACM, pp. 1725–1726, 2012. [Online]. Available: https://doi.org/10.1145/2245276.2232055

[43] F. Santos, C. Cappelli, F. Santoro, J. do Prado Leite, and T. Batista, "Aspect-oriented business process modeling: analyzing open issues," *Business Process Management J.*, vol. 18, no. 6, pp. 964–991, 2012. [Online]. Available: https://doi.org/10.1108/14637151211283375

[44] A. Jalali, "Assessing aspect oriented approaches in business process management," in *Perspectives in Business Informatics Research.* Lecture Notes in Business Information Processing, Springer, vol. 194, pp. 231–245, 2014. [Online]. Available: https://doi.org/10.1007/978-3-319-11370-8_17

[45] K. Yamalidou, J. Moody, M. Lemmon, and P. Antsaklis, "Feedback control of petri nets based on place invariants," *Automatica*, vol. 32, no. 1, pp. 15–28, 1996. [Online]. Available: https://doi.org/10.1016/0005-1098(95)00103-4

[46] W. Reisig, "Petri nets: an introduction, volume 4 of eatcs monographs on theoretical computer science," 1985.

[47] T. Murata, "Petri nets: Properties, analysis and applications," *Proceedings of the IEEE*, vol. 77, no. 4, pp. 541–580, 1989. [Online]. Available: https://doi.org/10.1109/5.24143

[48] H. M. Verbeek, T. Basten, and W. M. P. van der Aalst, "Diagnosing workflow processes using woflan," *The computer journal*, vol. 44, no. 4, pp. 246–279, 2001. [Online]. Available: https://doi.org/10.1093/comjnl/44.4.246

[49] I. Bider and A. Jalali, "Agile business process development: why, how and when-applying Nonaka's theory of knowledge transformation to business process development," *Information Systems and e-Business Management*, vol. 14, no. 4, pp. 693–731, 2016. [Online]. Available: https://doi.org/10.1007/s10257-014-0256-1

[50] A. Jalali, "Aspect mining in business process management," in *Perspectives in Business Informatics Research*. LNBIP, Springer, vol. 194, pp. 246–260, 2014. [Online]. Available: https://doi.org/10.1007/978-3-319-11370-8_18

[51] B. C. P. Brandão, F. Santoro, and L. G. Azevedo, "Towards aspects identification in business process through process mining," in *Proceedings of the annual conference on Brazilian Symposium on Information Systems: Information Systems: A Computer Socio-Technical Perspective-Volume 1*. Brazilian Computer Society, p.99, 2015.

Appendix

This appendix describes the structure of the survey and other relevant contents to promote a better understanding of the experimental material.

Structure of the survey

Students could choose answers based on scales from *extremely likely* to *extremely unlikely*. Here is the range of the scale among which students could select one option.

- extremely likely
- quite likely
- slightly likely
- neither likely nor unlikely
- slightly unlikely
- quite unlikely
- extremely unlikely

These options (with the specified order) are mapped to an ordinal scale where the extremely likely option is mapped to 7 and extremely unlikely option is mapped to 1.

In the following sections, we describe the two sets of questions for measuring the *perceived usefulness* and the *perceived ease of use* of our approach.

Questions for measuring Perceived Usefulness

These are the questions used to measure the *perceived usefulness* of our approach. The questions are designed based on the technology acceptance model [26].

- A.1.1. Using AO-BPM in this experience enabled me to model business processes more quickly.
- A.1.2. Using AO-BPM in this experience improved my performance when modeling business processes.
- A.1.3. Using AO-BPM in this experience increased my productivity when modeling business processes.
- A.1.4. Using AO-BPM in this experience enhanced my effectiveness on modeling business processes.
- A.1.5. Using AO-BPM in this experience made it easier for me to model business processes.
- A.1.6. I found AO-BPM useful in this experience.

Questions for measuring Perceived Ease of Use

These are the questions used to measure the *perceived ease of use* of our approach. Also in this case, the questions are designed based on the technology acceptance model.

- A.2.1. Learning how to use AO-BPM was easy for me.
- A.2.2. I found it easy to get AO-BPM to do what I wanted to do.
- A.2.3. I found AO-BPM understandable and clear.
- A.2.4. I found AO-BPM flexible to be used in modeling business processes.
- A.2.5. It was easy for me to become skilful at using AO-BPM.
- A.2.6. I found AO-BPM easy to use.

Evaluation results in detail

Figure below shows the results of the evaluation in detail. In general, we received quite good feedback from participants for all measures. None of the median for perceived usefulness and easy to use is below 6 and 5 respectively. This indicates good perceived usefulness and ease of use in general. The feedbacks for measures A.2.3. and A.2.5. varies the most, which indicates that the approach is not perceived as easy as other measures in regards for learning.
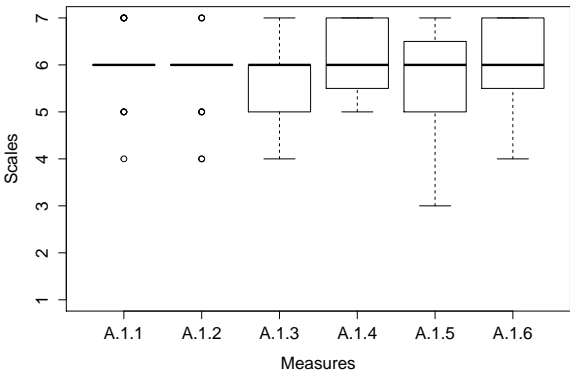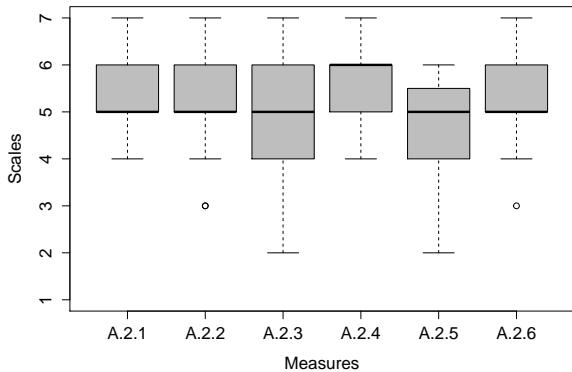
(a) Perceived Usefulness　　　　　　　(b) Perceived Ease of Use

Perceived Usefulness and Ease of Use of our approach in detail