Integrating non-functional security services in ADORE using multiple views modelling approaches

Fáber D. Giraldo  
System and Computer Engineering  
University of Quindío  
Armenia, Colombia  
email: fdgiraldo@uniquindio.edu.co

Raquel Anaya  
Informatics and Systems Department  
EAFIT University  
Medellín, Colombia  
email: ranaya@eafit.edu.co

Abstract—In this paper we propose the use of multi-view modeling approaches for managing the integration of conceptual artifacts. We propose the integration of behavioral and contextual approaches driven by aspects and models, in order to standardize and integrate behaviors derived from security requirements and functional specifications. Our goal is to build generic aspectual behaviors derived from security specifications and integrate them into models of business process workflows, using the Separation of Concerns principles. A Context is defined in terms of functional features and standards of the underlying services referenced by models. We use a multiple-view approach for modeling a concern derived from security specifications; posteriorly this concern is woven into a business process workflow, using aspectual composition principles, without mayor modifications to workflow itself.

Index Terms—Security Standards, Separation of Concerns, Business Processes Workflow, ADORE, XACML, Theme-UML.

I. INTRODUCTION

Separation of Concerns - SoC using aspect-based techniques have been widely accepted as a mechanism to encapsulate functional and nonfunctional features into reusable modules over a software solution. It can increase the level of abstraction by identifying common characteristics (which is a targeted goal of business process development). The quality attributes addressed in the design of a workflow may contain pre-defined standards, services and models. Therefore, the specification of the context of the quality attribute should be considered in terms of standards, service invocation and conceptual definition, for the subsequent design and integration into a business process workflow.

In most contemporary Service-Oriented Architecture (SOA) practices focused on separation of concerns, the properties related to quality attributes are specified and mapped in a set of services. This strategy requires that developers and SOA architects must configure properly the quality attributes in a range of services (usually every quality attribute covers multiple services simultaneously).

The use of SoC and aspects are extended to the treatment of quality attributes (as security and its derived implications, e.g., control access) so that business processes managed within a workflow consider additional features to functionality and behavior.

In a typical SOA context, there are several proposals for modelling business behavior (Activity diagrams in UML, BPMN and BPEL, specific proposals based on several symbols, and so on). Security (and quality attributes) are often handled at the infrastructure level and/or extensions. But it clearly impacts modeled behaviors. Unfortunately, approaches for business modelling (such as BPMN and BPEL) do not support the practice of formal separation of concerns per se [1]. As a result, proposals for integration of business processes behaviors, security operations and concerns result in a lot of extensions or new approaches no sufficiently applied.

The use of concerns managed by models allows flexible service orchestrations at a high level of abstraction, so that business processes are enriched with additional features [2] without representing major changes or changes to source code level.

In our previous works [3] we exposed how several works have been proposed for the derivation of security models from Business Processes Models (BPMN, BPEL and SOA models). So, from this idea, it is possible to deduce that i) security can be considered as a crosscutting concern, ii) security involves services, and iii) security could have a high-level support in a modelling context.

We think that principles derived from the AOSD and MDE provide a high degree of flexibility: AOSD can be applied to identify common concerns (such as security in our case), visualizing scenarios where they can be applied throughout the business process that is automated in a workflow. The business process models can be adapted to meet new requirements. Further changes to process models can be applied immediately to adjust business processes.
The core motivation of this work is to face the following research questions:

- Which is the way to apply and/or enrich the ADORE method (section II-A) in order to consider security based on standards?
- Which is the contribution of modeling languages based on UML for representing, at high abstraction level, the security as an aspect and the interventions of it over the core functionality?

This paper is organized as follows: the Section II briefly introduces the ADORE method for modeling service orchestrations, and the sample scenario used in our work. Section III presents the approach for introducing security as a concern, using an access control based approach; therefore, this concern could be woven in a business workflow. Section IV presents our multiple modelling view proposal for integration of services in a business process workflow. Section V relates previous works about integration of the security and business processes, and finally, section VII presents the conclusions.

II. ADORE AND ITS OPPORTUNITIES FOR IMPROVEMENT

A. The ADORE Method

The ADORE method (Activity moDel suppOrting oRches-tration Evolution)\(^1\) \cite{4,5,6}, defined by the MODALIS research group of University Nice-Sophie Antipolis\(^2\), provides a compositional technique for modeling complex service orchestrations. The models describe small orchestrations which are posteriorly woven to produce a model that represents an orchestration of a wide range of services\cite{6}. The models of small orchestrations, called fragments of orchestration, describing different aspects of a complex business process, where every aspect reflects a particular interest, may be a functional behavior or conduct directed by a quality attribute. Fragments are orchestrations not designed for direct implementation; the aim of the fragments is to be integrated into orchestrations. In ADORE terminology, a fragment is considered as an incomplete process.

B. A sample scenario of application of security services

The sample scenario on which this work was developed come from \cite{7}, who proposed a requirements specification document called Car Crash Crisis Management System that includes all the features of general systems of crisis management and some features for traffic accidents, such as facilitating the rescue of victims at the scene of the crisis, and the use of cranes for removing damaged vehicles.

The requirements specification document defines 10 use cases and 37 non-functional requirements for quality attributes such as Availability, Reliability, Persistence, Real-time, Security, Mobility, Statistic Logging, Multi-Access, Safety, Adaptability and Accuracy. Each scenario of the use cases defines a basic flow (successful scenario) which represents the normal flow of actions to handle a crisis. Subsequently, the use cases define a set of extensions to consider situations not covered by normal flows. The requirements specification document for the CCCms highlights the Security Requirement number 1 alludes to the definition of access control policies, where they describe the components and information that different users can add, access and update.

From the specification document of CCCms formulated by \cite{7} and developed under the ADORE method in \cite{5}, it highlights the proposal to address the concerns corresponding to non-functional requirements as fragments, so that they are connected to a larger service orchestration. Figure 1 presents an example of a fragment defined in ADORE for an use case of the CCCms.

However, when we look more closely this solution \cite{7} and the different application cases of the method (related in the official page of ADORE project\(^3\), we find opportunities to enrich the ADORE proposal in different ways:

1) Is evident the use of information of the modeled context expressed as variables that are introduced directly in the formulation of an ADORE fragment (e.g., see Figure 1). According with the ADORE method the knowledge of context mapped in the fragments is associated exclusively to the description of the process under consideration. An example is presented in the Figure 6 (a), where we show an ADORE fragment that contains information of the context in the variables block defined for this fragment. As it can be seen, the information about the context of the fragment (cmsEmployee, witness, crisisInformation and crisisCheckList variables) is defined by the ADORE expert from the specification of the use case. Therefore there is not a mapping of the features and context to the ADORE fragment. For example, Figure 2 presents a fragment that was woven in a workflow previously defined, without major information about the context of the fragment itself.

2) The specification of the security features in fragments was carried ad-hoc; we think the specification of the context of the quality attribute should be considered in terms of standards, service invocation and conceptual definition, for subsequent design and integration into a business process workflow. The conceptual specification of security attributes (and in general one quality attribute) using high level modeling, enrich the conceptual formulation of the ADORE fragments, so that it may address specialized behaviors derived from quality attributes at model level. Subsequently the quality fragments (such as security) could be woven in service orchestrations that compose complex business processes. The following sections are intended to explain the proposal to extend ADORE.

III. INTRODUCING SECURITY REQUIREMENTS IN ADORE USING AN ACCESS CONTROL APPROACH

Access control is a strategy for protecting system information. It is performed through the definition of roles associated to access permissions over specific resources. An access authorization is described explicitly through policies

\(^1\)http://www.adore-design.org/doku/

\(^2\)http://modalis.i3s.unice.fr/

\(^3\)http://www.adore-design.org/doku/examples/cccms/start
Fig. 1. Example of an **ADORE** fragment proposed by the **CCCms** sample scenario [5].

![Diagram of ADORE fragment](image)

\[
\text{fig1.png}
\]

Fig. 2. Example of weaving of **XACML** fragment with previous **CCCms** fragments defined in **ADORE** [3].

![Diagram of XACML weaving](image)

\[
\text{fig2.png}
\]
or permissions. The Role concept groups to users that meet a concrete profile for purposes of executing operations over one system.

In a preliminary work [3], we propose how an RBAC-XACML artifact, that belongs to the quality attribute of security under control access policies, can be considered as a concern (e.g., aspects weaving), and its application could be extended to a set of services, applications or business processes if it exists a mechanism for adding compositions of services in a particular business process and for considering security operations at the level of role-based control access policies. With this purpose, the RBAC-XACML concern was modeled as an ADORE fragment and it was woven into a previous ADORE model that represents the CCCms.

Our main conclusion of that preliminary work [3] is that management functionality and quality attributes (particularly security) may be performed using an aspect composition approach at higher levels of abstraction, where it is possible to define aspect based compositions under the same architectural principle or manager without using different technologies that support the same standard, or different modifications / standard extensions previously conceived.

In order to explain how the ADORE approach can integrate the security attribute-based standards as an aspect, we must first make a brief introduction to the standards that was considered (part III-A) and we propose an integration of respective metamodels (part B).

A. RBAC and XACML

The RBAC - (Role Based Access Control) model is a generalized approximation to access control where roles represent functions defined in an organization and permissions (defined as the ability or right to perform some action on a resource under certain conditions previously specified) depend on roles instead of users. The authorizations associated with a role are strictly related to data objects and resources necessary to perform the functions associated with the role. The users are simply allowed to play the appropriate role. The main benefit of adopting RBAC lies in simplifying the task of defining security policies for business experts who have no specific knowledge of software platforms to support security operations. The system management efforts are minimized due to the small number of relationships required to link roles with permissions.

The eXtensible Access Control Markup Language or XACML is an OASIS standard to specify and to force access control policies independent of any software platform. XACML (currently version 3.0) has become the de facto standard for specifying access control policies on web services. Since 2004, OASIS defined a XACML profile to RBAC, to link RBAC practical solutions in web services environments⁶. The administration of privileges in a distributed environment using solutions based on RBAC provides a greater control of these, also reducing the complexity involved in this management process. The XACML-RBAC profile makes possible the management of security policies in a distributed environment because of the location of roles and policies from distributed sites that can be managed independently [8].

XACML defines requirements for access control to protected resources of an application. The language of access decision XACML is used to represent a query that asks whether a particular action should be allowed on a resource. If a policy associated with the resource accessed is found, the attributes in the request are compared with the attributes listed in the policy rules. Finally, it sets a response which determines whether the request should be allowed or not, using one of four values: Enabled (Permit), Denied (Deny), Undetermined (Indeterminate - failed or missed a required value, so that you cannot make a decision) or Not Applicable (the request cannot be responded by a service).

Figure 3 shows the specification model of XACML published by OASIS. As such, OASIS does not use the term metamodel to refer to this model, however in this work shall be considered as metamodel because it complies with the MOF to the model specification to high level of abstraction.

B. Integrating of ADORE metamodel with XACML and SecureUML metamodels

In [4] a set of ADORE MOF metamodels are proposed for defining the BusinessProcess concept as a set of variables, activities and its order relations. Also, these metamodels define other important elements in ADORE, such Orchestration concept (realization of the behavior associated to an operation of service), the Fragment concept (incomplete behavior which aims to be integrated into another process), and the Activity concept (use of input variables and assignment of results to output variables) with its respective types (invocation, variable assignment, fault reporting, message reception, response sending and synchronization activity - Nop).

Many works related with security in MDE contexts include the SecureUML proposal as one of the most representative alternatives to modeling security based on UML[9]. SecureUML is a MDE approach for specifying access control requirements based on roles over software systems, using metamodeling techniques to support RBAC with restrictions of authorization specified in Object Constrain Language - OCL. Several authors have suggested considering together SecureUML and XACML so that the models reflect conditions of access control in SecureUML that are subsequently mapped to XACML, making even some level of automation to generate transformation rules from UML-SecureUML models to XACML rules[10][11].

In [12] the authors expose that SecureUML does not provide any mechanism for specifying the strategies for resolving conflicts between policies, while in XACML this resolution is clear and concrete. Additionally, SecureUML does not support the grouping of policies.

⁵Available in http://docs.oasis-open.org/xacml/2.0/XACML-2.0-OS-ALL.zip
5

From the ADORE scheme presented in [4] and considering that the SecureUML metamodel requires the use of an additional language to support the model and validation of access controls, we propose to integrate of ADORE, XACML and SecureUML metamodels in order to establish, from a high level of abstraction, the specifications of an ADORE fragment responsible for addressing the management of role-based security and access control.

The ADORE method is supported by a metamodel using the MOF OMG standard. Given that the ADORE natively does not involve security, the formulation at MOF level of ADORE fragments does not incorporate behavioral elements derived from quality attributes (such as access control for the specific case of security), this implies that is necessary an integration at level of MOF metamodels to conceptually support the definition of this kind of fragments; enriching the ADORE method in itself.

Figure 4 shows the integration of these metamodels, this integration defines an EClass in EMF called Secure XACML Fragment, which inherits all the properties of an ADORE fragment (at business process levels and activity levels), and will have to hook, predecessors and successors such as an ADORE common fragment. Additionally, to the concept Secure XACML Fragment added the relationships of composition with elements defined in the XACML metamodel to specify that an ADORE fragment of access control security will have: i) a subject who initiates the query of access on a given resource, ii) the resource on which access is being requested, iii) the policy or set of policies related to resource.

To ensure the traceability of the elements XACML with the requirements of access control security specified for the workflow of business processes, we define a relationship mapping (maps) between elements of the XACML metamodel and elements of SecureUML metamodel; the goal with this mapping relationship is to establish that any requirement of role-based security and access control defined for a workflow, can be immediately mapped to XACML elements managed by an aspectual ADORE fragment. Ideally, the XACML access control rules should be derived from security models. Therefore, the metamodel should reflect the mapping of XACML to an specific concepts of SecureUML in order to ensure that the components involved in access control are derived from the early stages of design and modeling of the system.

IV. INTEGRATION OF SERVICES WITHIN BUSINESS PROCESS WORKFLOWS THROUGH MULTIPLE VIEW MODELING

The design of systems such as business process workflows usually tackles multiple views for considering concerns derived from several dimensions (functionality, quality, business, data, technology, among others). Concerns are composed or woven for building a solution. In [4], it is highlighted how the ADORE method supports the composition of entities driven by behaviours (fragments), e.g., units of ADORE modeling expressing the concerns of multiple views in terms of behavior. These previous demos and examples developed in ADORE model contain behaviours without an explicit distinction of the structural dimension that supports these behavior. As we showed in Figure 6, the definition of fragments under ADORE induced to structural information that is embedded directly in the behavior that is being specified.

Complementing the definition of fragments under the ADORE method is an alternative for combining behaviours (ADORE) and structures that represent operation contexts. The use of model-driven approaches allow the definition of a generic fragment to be considered from early stages of building the business process using high abstraction levels. Ideally, we should model the context of the operation to be encapsulated in an ADORE fragment (structural information). This context can be composed from the specification of the operation, according to any standard or referent used (such...
XACML), and a set of features of workflow under consideration.

For the purpose of building fragments capable of addressing non-functional behaviors and flexible enough to weave with different base functionalities, we propose a multiple view approach: the functional-behavior perspective supported by the ADORE method and a structural perspective supported by Theme/UML\(^8\). These proposals are based on works such as [13][14][15]. We have decided to use a subset of the diagrams of the modeling phase established in the Model-Driven Theme/UML process development to show the mapping of the information of the context of the quality attribute towards the ADORE fragment.

In the functional perspective, it is necessary to complement the definition of fragments under the ADORE method, in order to specify the security behavior so parameterizable so that these generic fragments can be articulated into different applications.

The ADORE security fragment for access control based on XACML is in itself a Policy Enforcement Point (PEP), according to the XACML request/response process defined in Section III-A. This PEP captures the request for access by the user, performs the invocation of the PDP responsible for obtaining the policies related to a request, invokes the PIP to consult the attributes of the policy found by the PDP, and then compare these attributes with the attributes of the access request for proceeding to allow or deny access. It must be remembered that the code presented by the fragment is committed in services explicitly invoked in the fragment, so the subsequent implementation XACML process access control is under the responsibility of security experts.

In this sense, in the Figure 8, we propose a representation of higher abstraction level of the security behavior; according with XACML standard. Previously (Figure 2), this fragment was woven in a predefined business process workflow for the CCCms sample scenario.

In the structural perspective we need to represent the structural entities in order to combine behaviors (ADORE) and structures that represent operation contexts. This context is composed from the specification of the operation according to any standard or referent used (such XACML), and a set of features of workflow under consideration (see Figure 7).

The context is defined in terms of the features specified for the system, the set of standards that govern the application and the invocation of underlying services, as the XACML standard for this case.

Figure 5 presents a proposal to justify the specifications of generic ADORE fragments that posteriorly will be applied in different business process workflows. The main goal is to integrate the behavior management approach of the ADORE method with an approach of multiple-view modeling used to support the structural information that defines the concerns of quality attributes (fragment context). Then, the concerns will be woven with fragments of business processes using composition mechanisms of ADORE. The crosscutting behaviors modeled in Theme/UML are woven with other aspects

\(^8\)http://www.dsg.cs.tcd.ie/aspects/themeUML
or fragments through the **book**() activity natively defined in **ADORE** and exposed in the metamodel of Figure 5.

Figure 7 presents an application of the proposal of integration of the modeling approaches for behaviours and multiple view, for formulating an **ADORE** fragment that represents the control access security process defined under XACML standard defined in the section III-A.

Therefore, it is possible to restrict the formulated fragments to a functional dimension according with the conception of behavior belonging to the **ADORE** method.

The consideration of the conceptual specification of quality attributes behaviors would enrich the conceptual formulation of the **ADORE** fragments, so that it may address specialized behaviors derived from quality attributes, at model level, which could subsequently be woven in service orchestrations that compose complex business processes. Composition of services has received much interest to support business-to-business or enterprise application integration [16].

The behavior perspective characteristic of the **ADORE** method leads to the definition of functional entities that do not consider independent views derived from non-functional behaviours, which previously defined a conceptual structure. The definition of fragments under **ADORE** induced to structural information is embedded directly in the behavior that is being specified. **ADORE** has not a multiple view approach: fragments are defined in an only view (functional view in textual model) so that the structure is immersed in this model. This work proposes the use of a functional perspective supported by the **ADORE** method and a structural perspective supported by Theme/UML\(^9\), for the purpose of building fragments capable of addressing non-functional behaviours.

Such as was exposed in section III-A with the XACML standard, a quality attribute could contain a set of associated services and structures or specific dimensions, as hardware/software technologies that support security operations. Other types of concerns address specifications of dimension, e.g., if a business process model must consider the managing of business rules, the fragment or fragment set must consider specific operations over a business rule engine.

Additionally in the **ADORE** fragments built for the different application cases of the method (related in te oficial page of **ADORE** project\(^10\)), is evident the use of information of the modeled context expressed as variables that are introduced directly in the formulation of a fragment. According with the **ADORE** method, the knowledge of context mapped in the fragments is associated exclusively to the description of the process aborded. An example is presented in the Figure 6, where we show an **ADORE** fragment that contains information of the context in the variables block defined for this fragment. As it can be seen, the information about the context of the fragment (\(\text{cmsEmployee, witness, crisisInformation and crisisCheckList}\) variables) is defined by the **ADORE** expert from the specification of the use case. Therefore, there is not a mapping of the features and context to the **ADORE** fragment.

The formulation of the fragment shown in Figure 7 makes evident how the fragment is defined in terms of a generic solution (the fragment is modeled from the XACML standard) but not from the problem itself, as it is exposed in the **ADORE** fragment of Figure 6. The incorporation of Theme/UML helps to define the aspectual fragment as a generic model from the structural and behavioral perspectives.

The whole XACML access control process, exposed in section III-A, is modeled in an **ADORE** fragment due to the XACML process that is based on the invocation of underlying services that implement security operations. So, a fragment can be further woven with other fragments and larger orchestrations of services to adapt to the business process, so it is possible to adapt the workflow by aspectual composition and the addition of this control access fragment. Figure 8 presents the **ADORE** fragment for encapsulating all XACML process in one unit of modeling of the business process. Figure 2 shows the weaving of XACML fragment with others **ADORE** models previously defined for the CCCMs (the \(\text{execResMission}\) orchestration).

**V. Related works**

Previous proposals of modeling approaches integration for definition of services in early levels of software construction are been identified. For example, in [17] authors propose a reuse approach that applies weaving of generic and target modules expressed in a single modeling paradigm (OPM - Object-Process Methodology) that combines structure and behavior by presenting and linking objects and processes (including aspect features) in the same single diagram type. In [18] it is proposed an extension of UML named UML-S (UML for Services) for modeling web services as well as their interactions. Its extension generates WS-BPEL 2.0 code from UML-S diagrams. Granell et al in [19] formulate an approach for facilitating the reuse of existing compositions to create new complex compositions. Their proposal provide a service model represented jointly in terms of reusable services, building blocks for constructing and reusing new compositions, and abstract patterns for managing reusable service composition. In [20] authors proposed a holistic approach to engineering service compositions, abstracting formal behavioral models from interacting service orchestrations, their choreography policies, and deployment architecture scenarios. However, the above proposals do not consider services derived from quality attributes. Also, most of these proposals are driven by object oriented principles, so that, these proposals must be represented through stereotyped classes or other UML symbols in order to manage and incorporate aspectual modules belonging to ThemeUML and **ADORE**. It represent additional effort with respect to native modeling approach proper of ThemeUML and **ADORE**.

**VI. Conclusions and further work**

This work has formulated a proposal for integration of MDE and AOSD principles with the purpose of establishing mechanisms of adaptation of business process workflows at modeling level. Our proposal incorporate security constraints

---

\(^9\)http://www.dsg.cs.tcd.ie/aspects/themeUML

\(^10\)http://www.adore-design.org/doku/examples/cccms/start
Fig. 5. Integration of Theme/UML - ADORE proposal.

Fig. 6. Example of information of context in an ADORE fragment.

Fig. 7. Using Theme/UML for defining an ADORE fragment to support XACML standard.
based on access control rules defined by the RBAC model and the XACML standard. In this way, workflows are adapted since design levels. The proposal uses intensively the ADORE method to specify both the business process as well as the addressed security considerations, which are added to the workflow by aspectual composition.

The proposal has specified a compositional element called orchestration fragment according to the ADORE method. This fragment was built under the conceptual principles established in the XACML model (section III-A) to support security operations based on access control of roles. Subsequently, the fragment is added or woven into a sample scenario. This work proposes the use of a multiple-views approach based on Theme/UML for formulating ADORE fragments with generic behaviors derived from specifications of quality attributes expressed as services invocations (section IV). We propose the use of Theme/UML for managing the information related to the context of ADORE fragments applied to the formulation of generic behaviours. The representation of context, besides using fragments, could consider the application of Theme/UML for deriving rules and transformations that contribute to control the repetitions of services previously invoked.

We have shown how the design of fragments of non-functional requirements could present invocations established from the pre-defined standards and models, and ideally it should be considered in the design of aspectual encapsulation of the attribute using ADORE fragments. Therefore, the specification of the context of the quality attribute should be considered in terms of standards, service invocation and conceptual definition, for subsequent design and integration into a business process workflow. Also, we showed how we use modelling proposals based on UML for addressing the conceptual specification of quality attributes behaviors, which would enrich the conceptual formulation of the ADORE fragments. In this way, it is possible to address specialized behaviors and its structural properties derived from quality attributes, at model level; and subsequently, it can be woven in service orchestrations that compose complex business processes.

Our work is in progress and it needs to be validate through formal experiments. However, we visualize some further works derived from our integration proposal:

A. Explicit integration of behaviors derived from quality attributes in ADORE

The formulation of fragments derived from security operations (access control) allows to deduce the possibility of defining fragments or orchestrations of quality attributes behaviors, in a general way. Such fragments could be reused in several business processes workflows, due to the separation of the underlying service invocations grouped by fragments of specific behaviors, and the business process managed by workflow. This future work contemplates i) the definition at the ADORE metamodel level of the representative behaviors of quality attributes with direct impact on the execution of business processes workflows (for example security operations such as integrity, authentication and privacy), and ii) The implementation by ADORE fragments of a library of security reusable behaviors in different workflows.

Currently, the structure of the XML file generated from ADORE to represent compositions between fragments and orchestrations, does not contain XML tags to identify if one fragment is derived from a quality attribute. This tag should be added from the same ADORE engine, previous conceptual redefinition of ADORE metamodel.
B. Incorporating control fragments

The invocation of aspectual fragments responsible for validating access control rules can generate a processing overload, because the invocations of the services involved in a process based on XACML (PEP, PDP and PIP) could be repeated to validate the access of a previously evaluated role. Therefore, we propose the incorporation of control fragments responsible of preventing the repetition of operations derived from quality attributes that may affect seriously the behavior of the workflow under execution. These fragments should use the information of the execution context of the workflow itself. For this reason, it is necessary to strengthen the representation of the execution context through Theme/UML. Thus, there is a single instance that influences the operations of fragments weaving in a process of dynamic adaptation of workflows. Through the aspectual guidance of fragments in ADORE method it is possible to consider a specialized fragment of information to manage a business process context, it could be woven in any fragment/orchestration that requires the validation of the previous invocations of a similar operation.

C. Integration of aspectual approaches with our proposal

Any aspectual approach that manages a specific part of the development process could be candidate to be combined with the ADORE method and its respective artifacts at model levels; for example, the EARLY ASPECTS approach exposed in [21]. In order to validate the ADORE method respect to EARLY ASPECTS approaches (for example) it is necessary to execute a future work of automatization of transformation defined in Theme/UML, using a profile approach for generating the code of fragments/orchestration defined in the ADORE model.

ACKNOWLEDGMENTS

F.G. thanks Prs Mireille Blay-Fornarino (University of Nice-Sophia Antipolis, France) and Sébastien Mosser (University of Lille I, France) for their collaborations and their valuable feedback.

REFERENCES


