

# Hierarchical Adaptation of Workflows Defined in Ontologies

Alvaro E. Prieto, Adolfo Lozano-Tello

Quercus Software Engineering Group, University of Extremadura, Cáceres, Spain  
{aeprieto, alozano}@unex.es

**Abstract.** Administrative processes are a type of business process frequently used in hierarchical organizations. The workflows of these processes are often defined generically in the level of management or governance of the organization. These workflows must be specialized in the lower levels of the organization hierarchy before they can be used in them. In addition, any change in the generic workflows must be also propagated to the lower levels. Traditional approaches of workflow specialization do not provide a complete solution to these issues. However, some strategies of ontology evolution can be adapted to solve these problems. In this paper we propose a method that combines approaches of both research areas called Workflow Hierarchical Adaptation Method. This method can be applied to workflows defined in ontologies using the WEAPON Model.

Keywords: hierarchical adaptation; business process, administrative process, workflow, ontology, WEAPON.

## 1 Introduction

Public institutions and large companies formally define their business models on administrative processes. These processes are often governed by laws, regulations or well-defined action rules. They are usually characterized by the submission of an application form that will be handled by different users at different stages. These users may access to the current information of the process in order to provide data that will be included in its dossier. All these information will be used by the person responsible for taking a final decision on the application of each process. A request for consumer arbitration, a credit application or a holiday application are examples of this type of processes

Administrative processes, as most business processes, can be automated using Workflow Management Systems (hereinafter referred to as WfMS). But the definition of workflows of administrative processes is not a trivial issue. This is due to they are often defined generically in the level of management or governance of the organizations. However, these workflows must be specialized before they can be used in the lower levels of the hierarchy of dependent institutions, subdepartments, delegations, etc. As a result, the engineer that defines the workflows of administrative

processes faces two problems. First, the specialization of the workflow definition to the particular conditions of the hierarchy level where it will be used without loss of the restrictions established in the generic definition. Second, the management of changes in the workflow definitions when the laws or regulations that govern the administrative process are changed. This second issue also includes the propagation of these changes to the specialized workflow definitions in lower levels.

Typically, traditional approaches of inheritance or specialization of workflows focus only on the process perspective, i.e., on the routing and control flow aspects [1]. However, in many cases the data managed and the workflow participants are also affected by the specialization or the management of changes in administrative processes. Furthermore, these approaches neither deal with the problem of propagating changes among levels.

On the other hand, ontologies are actually used to define the knowledge of a domain in a complete, precise and reusable representation which is both machine-readable and human-understandable [2]. Ontologies help to impose restrictions on classes and attributes that allow to keep integrity and consistency of the representation when the structure and the characteristics of data are changed. The management and propagation of changes among ontologies is a research area known as Ontology Evolution.

Using as a basis the advances in workflow specialization and ontology evolution, in this paper we propose a method that joins together the specialization of workflow definitions, including data and users, and the management and propagation of changes among these definitions. This method is called Workflow Hierarchical Adaptation Method. This method has been designed to be applied to workflows defined using the WEAPON model [3]. WEAPON provides a meta ontology (OntoMetaWorkflow) as a basis for defining workflows of administrative processes together with the methods and tools that support the definition and management of workflows using this meta ontology. WEAPON model is a complete restructuring of the WfMS model based on ontologies that was proposed in [4].

This paper is structured as follows: section 2 identifies related work and section 3 describes the Workflow Hierarchical Adaptation Method.

## **2 Related Work**

There are many interesting approaches in the three research areas that have helped us to elaborate our proposal. It is impossible to enumerate each one in this section. We have selected some important papers in the three domains.

### **2.1 Use of Ontologies in WfMS**

Important approaches in the application of ontologies to WfMS were presented in [4], of which we can highlight the one of Vieira et al. [5] that proposes a solution to make workflow execution more flexible and also interesting is the work of Gasevic et al. [6] which provides a Petri net ontology. We can also mention the recent survey about

Semantic Business Process Management is available in [7] with more examples of integration of both fields.

## **2.2 Inheritance or Specialization of Workflows**

The approach of Van der Aalst et al. [1] is, possibly the best known approach in inheritance of workflows. They apply the concepts of encapsulation and abstraction of object oriented paradigm to workflow. Also interesting is the approach of Wyner et al [8] that is based on the view that specialization is a subsumption. More approaches are referenced in a recent survey [9].

## **2.3 Ontology Evolution**

Stojanovic et al. [10] propose a six phases ontology evolution process: capturing, representation, semantics, implementation, propagation and validation. Plesser et al. [11] proposes a five phases process: request, implementation, detection, recovery and propagation. De Leenheer et al. [12] describes an ontology evolution process composed of four activities (request, accepted, implement, verify and validate) over three phases (initiation, execution and evaluation). Recent surveys about ontology evolution are available in [12,13].

# **3 Workflow Hierarchical Adaptation Method**

The Workflow Hierarchical Adaptation Method defines the processes, operations and logs needed to specialize workflows of administrative processes and manage their changes when these workflows are represented in ontologies following the WEAPON model. This method has been inspired by the ideas described in [1,8,10-12].

Firstly, the method is composed of two processes: specialization and evolution. The first process focuses on specializing the data, users and activities of a generic workflow definition to the particular characteristics of a hierarchy level. The second one includes the management of changes that may occur in higher level definitions and their propagation to the lower level definitions. Both processes have been defined with the aim of keeping the consistency among the generic and specialized workflows. Figure 1 shows an overview of the use of these processes.

Secondly, it has also been defined a set of operations that can be applied to the elements of OntoDD and OntoWF of the WEAPON Model (OntoDD contains the taxonomy of data which will be used in the corresponding domain and the taxonomy of the possible workflow participants and OntoWF contains the concrete workflow of the administrative process, including its properties, the activities that it contains, the order of execution of said activities). These operations are divided into specialization and evolution operations. Each operation has associated a group of restrictions and implications. The first ones must be checked before applying the operation and the second ones contains a list of the side effects that involves performing the operation.

Lastly, two logs are necessary in the method in order to store the operations performed applying processes, a specialization log and an evolution log. A brief description of the phases of each process is presented in next subsections. Figure 2 shows a schema of these phases.

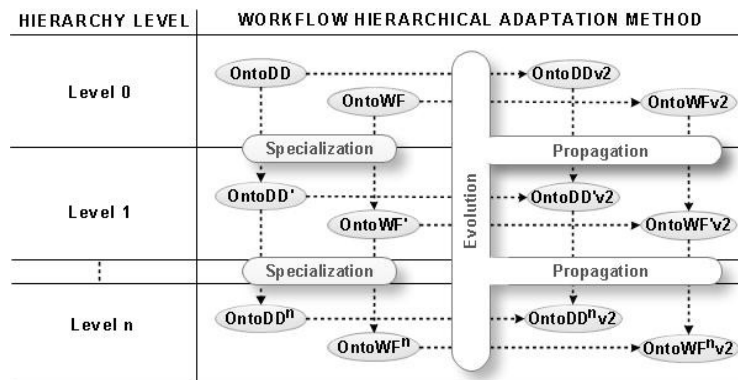


Fig. 1. Overview of Workflow Hierarchical Adaptation Method.

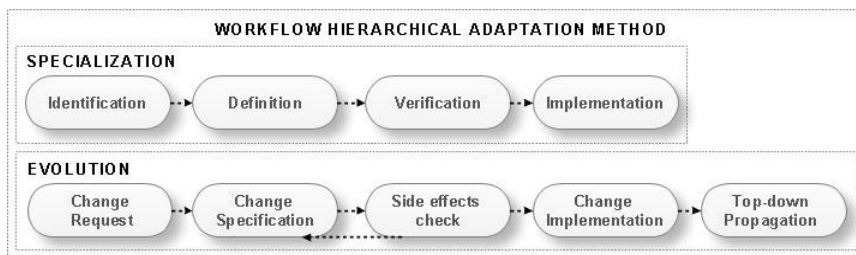


Fig. 2. Phases of the processes.

### 3.1 Phases of Specialization Process

The specialization process is composed of four phases:

1. Identification: the process starts with the identification of the specialization requirements of the next level in the hierarchy of the organization.
2. Definition: in this phase, the process engineer must establish what specialization operations need to be applied to accomplish with the specialization requirements.
3. Verification: in this phase the process engineer must analyze if some of the specialization operations specified can produces inconsistencies among the generic ontology and the future ontology. To support this phase, each specialization operation provides detailed insight into restrictions and implications of each operation being performed.
4. Implementation: the selected specialization operations are applied to OntoDD and OntoWF in this phase. In addition, these operations must be stored in the specialization log.

### 3.2 Phases of Evolution Process

The evolution process is composed of five phases:

1. Change Request: in this phase, the process engineer captures changes that must be applied to the workflow. These changes are usually forced by alterations in the laws, policies or rules that govern the administrative processes.
2. Change Specification: in this phase, the changes are represented using some of the change operations that can be applied to OntoDD or OntoWF. The process engineer must verify that the restrictions of the operation are fulfilled before choosing it.
3. Side effects check: in this phase, the process engineer, following the implications associated to every change operation, must elaborate two lists: one with the implications to the OntoDD and OntoWF of the same level and one with the implications to OntoDD and OntoWF of lower levels. Every implication of the first list must be analyzed in order to determine whether involves new change requests or not. If so, it will be necessary to come back to the second phase and manage these new changes. The implications of the second list will be used in the Top-Down Propagation phase.
4. Change implementation: the specified changes are applied to OntoDD and OntoWF in this phase. In addition, all applied changes must be stored in the evolution log.
5. Top-Down Propagation: in this phase it is necessary to apply recursively the ontology evolution process to OntoDD and OntoWF of the lower levels. In this case, the list of implications of lower levels obtained in the Side effects check phase are the changes that must be applied to the workflow.

## 4 Conclusions

In this paper we have presented the Workflow Hierarchical Adaptation Method based on ontologies. The Workflow Hierarchical Adaptation Method identifies the activities and steps to be followed to specialize and manage changes of workflows of administrative processes defined using the WEAPON model. This method provides a solution to the problem of specialization of the workflow definition of administrative processes within the hierarchy of an organization. It also provides a proposal to the problem of propagation of changes from workflow definition in higher levels to lower levels. Unlike traditional approaches, the proposed method includes the workflow data and the workflow participants and not only the activities and their control flow.

**Acknowledgments.** This work has been developed under support of Ministerio de Ciencia e Innovacion Project (TIN2008-02985), FEDER, Junta de Extremadura and Plan de Iniciacion a la Investigacion, Desarrollo Tecnologico e Innovacion 2010 de la Universidad de Extremadura (ACCVII-04).

## References

1. W.M.P.V.D. Aalst and T. Basten, "Inheritance of workflows: an approach to tackling problems related to change," *Theoretical Computer Science*, vol. 270, 2002, pp. 125-203.
2. A. Gómez-Pérez, M. Fernández-López, and O. Corcho, *Ontological Engineering. With Examples from the Areas of Knowledge Management, e-Commerce and the Semantic Web*, London: Springer-Verlag, 2004.
3. Á.E. Prieto and A. Lozano-Tello, "WEAPON: Modelo de Workflow con Ontologías para Procesos Administrativos," *Actas de las XIV Jornadas de Ingeniería del Software y Bases de Datos*, A. Vallecillo and G. Sagardui, eds., San Sebastián: 2009, pp. 291-302.
4. Á.E. Prieto and A. Lozano-Tello, "Use of Ontologies as Representation Support of Workflows Oriented to Administrative Management," *Journal of Network and Systems Management*, vol. 17, 2009, pp. 309-325.
5. T.A.S.C. Vieira, M.A. Casanova, and L.G. Ferrao, "An Ontology-Driven Architecture for Flexible Workflow Execution," *Proceedings of the WebMedia & LA-Web 2004 Joint Conference 10th Brazilian Symposium on Multimedia and the Web 2nd Latin American Web Congress*, Los Alamitos, CA, USA: IEEE Computer Society, 2004, pp. 70-77.
6. D. Gasevic and V. Devedzic, "Petri net ontology," *Knowledge-Based Systems*, vol. 19, Aug. 2006, pp. 220-234.
7. H. Hoang, P.-C. Tran, and T. Le, "State of the Art of Semantic Business Process Management: An Investigation on Approaches for Business-to-Business Integration," *Intelligent Information and Database Systems*, N.T. Nguyen, M.T. Le, and J. Świątek, eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 154-165.
8. G. Wyner and J. Lee, "Applying Specialization to Petri Nets: Implications for Workflow Design," *Business Process Management Workshops. BPM 2005 International Workshops, BPI, BPD, ENEI, BPRM, WSCOBPM, BPS*, Nancy, France, September 5, 2005. Revised Selected Papers, C.J. Bussler and A. Haller, eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, pp. 432-443.
9. H. Schonenberg, R. Mans, N. Russell, N. Mulyar, and W.V.D. Aalst, "Process Flexibility: A Survey of Contemporary Approaches," *Advances in Enterprise Engineering I*, W. van der Aalst, J. Mylopoulos, N.M. Sadeh, M.J. Shaw, C. Szyperski, J.L.G. Dietz, A. Albani, and J. Barjis, eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 16-30.
10. L. Stojanovic, A. Maedche, B. Motik, and N. Stojanovic, "User-Driven Ontology Evolution Management," *Knowledge Engineering and Knowledge Management: Ontologies and the Semantic Web*, A. Gómez-Pérez and V.R. Benjamins, eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2002, pp. 133-140.
11. P. Plessers and O. De Troyer, "Ontology Change Detection Using a Version Log," *The Semantic Web – ISWC 2005*, Y. Gil, E. Motta, V.R. Benjamins, and M.A. Musen, eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, pp. 578-592.
12. P. De Leenheer and T. Mens, "Ontology Evolution. State of the Art and Future Directions," *Ontology Management for the Semantic Web, Semantic Web Services, and Business Application*, M. Hepp, P. De Leenheer, A. de de Moor, and Y. Sure, eds., New York: Springer, 2008, pp. 131-177.
13. G. Flouris, D. Manakanatas, H. Kondylakis, D. Plexourakis, and G. Antoniu, "Ontology change: classification and survey," *The Knowledge Engineering Review*, vol. 23, Jun. 2008, pp. 117-152