

---

# An Approach for Building an OWL Ontology for Workflow Interoperability

S.Hamri<sup>1</sup>, N.Boudjlida<sup>2</sup> and M.Boufaida<sup>1</sup>

<sup>1</sup> Laboratory LIRE, Computer Science Department, Mentouri University of Constantine, Algeria, 25000. hamris@yahoo.fr, boufaida\_mahmoud@yahoo.fr

<sup>2</sup> University of Nancy 1, LORIA. UHP, France. Nacer.Boudjlida@loria.fr

**Keywords:** interoperability, workflow, MDA, ontology, OWL

## 1 Introduction

Most explorations of interoperability have been made in different domains: data bases, engineering software, Workflow systems [3], Web services composition / orchestration [4], [5], [6], [2] in the past few years. In the field of Workflow interoperability (or business process), the WfMC (Workflow Management Coalition) has defined a canonical model [7] called XPD (XML - Process Language Definition) as a language for process interchange. The Business Process Management (BPM) has mainly focused on Web service technology and has come up with a multitude of Web service composition languages standards such as: ebXML [2], XLANG [5], WSFL (Web Services Flow Language) [4] WSCI (Web Services Choreography Interface) [6], BPML (Business Process Modeling language) [4], and other languages for orchestrating business process activities in the Web. All these languages are based on XML and unfortunately, no common standard or no common meta-model has been agreed upon. Hence, there is a need for the semantics of concepts. As we consider a Workflow system as a support for business activities, it is necessary to take into account the knowledge context of these activities. The use of ontologies is one mean to consider this kind of knowledge. Indeed, ontologies facilitate interoperability, by aligning different terms that might be used in different Workflow models. Accordingly, in this paper, we propose an ontology-based approach for building an OWL ontology for Workflow interoperability. This latter constitutes then, a common ontology that makes Workflow models able to understand each other.

## 2 Overview of the Approach

The proposed approach uses the MDA architecture, on the one hand, for building a Workflow meta-model using MOF (Meta-Object Facility) [13] and to exploit ontologies for defining the concepts that describe the knowledge of the Workflow domain, on the other hand. In fact, this combination so-called integration of approaches has been used in many research works for a while [9], [11]. This architecture is an MDA-defined ontology architecture that includes an Ontology Definition Meta-model (ODM) [9], and an OWL Meta-model. ODM was designed to enclose common ontology concepts with using MOF. Its construction is based on OWL (Ontology Web Language) [10]. The process for building our OWL ontology Workflow is defined by four steps (Fig 1).

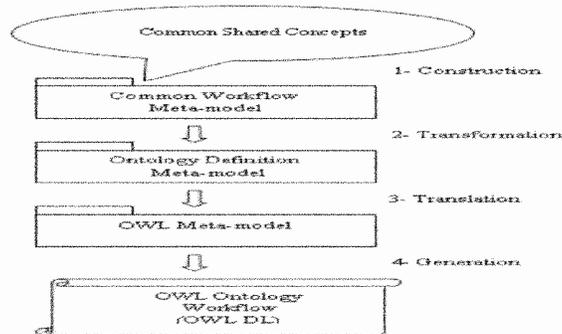


Fig. 1. Steps for building an OWL ontology Workflow.

The first step is to get all common concepts and to convert them into MOF-classes by making associations between them. For the transformation and the translation steps, we use the corresponding table of mappings (see table 1) between MOF concepts, ODM concepts and OWL concepts. Finally, the last step use the ontology tool Protégé [12] for generating our OWL ontology Workflow.

## 3 The Common Workflow Meta-model

To build the common meta-model, we have investigated the concepts that are shared between all Workflow models. We have firstly studied the most modeling languages used in our domain. Next, we have extracted a core of basic concepts that is common to the set of Workflow models. Then, we have compared the used concepts and have aligned them up according to their objectives as defined by their designers. These extracted concepts would be thus, generic and re-usable. We call this set of concepts and their semantics the common Workflow meta-model [1]. As the notion of meta-model is strongly related to the notion of ontology [8], this common meta-model (Fig 2) is considered as a common ontology in the field of software engineering.

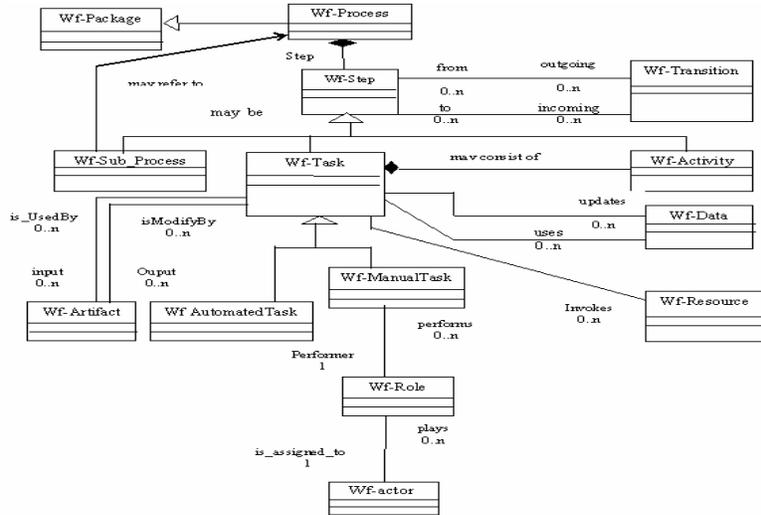


Fig. 2. A common Workflow meta-model.

#### 4 Mapping Common Meta-model to OWL Meta-model via ODM

Once the common meta-model built, it is translated into ODM and subsequently, from ODM to OWL meta-model using the bellow table.

Table 1. Mappings summary between MOF concepts, ODM concepts and OWL concepts.

MOF Concepts	ODM Concepts	OWL Concepts
Package	Class Ontology	OWL: Ontology
Class	Class Class	OWL: Class
Attribute	Class DatatypeProperty, if the type of Attribute is related to Data Values	OWL: DatatypeProperty, if the type of property is related to Data Values
Attribute	Class ObjectProperty, if the type of Attribute is related to Classes	OWL: ObjectProperty, if the type of property is related to Classes
Association	Class ObjectProperty	OWL:ObjectProperty
Multiplicity	Class Restriction, Class Cardinality, Class MinCardinality, Class MaxCardinality	OWL:Restriction, OWL:Cardinality, OWL:MinCardinality, OWL:MaxCardinality

Finally, we mainly focus on OWL DL (Description language) of Protégé OWL Plugin [12] for the generation of our OWL ontology. This approach via ODM is open and flexible since when one wants to support a new language (e.g., DAML+OIL), we only use the ODM-based principle. Furthermore, it enables to build an architecture whose advantages are: opening, flexibility, and evolution since new Workflow models can easily join to the framework of interoperability. So, if some provides of Workflow models want to interoperate with their own concepts, alternative methods are used for solving the problem of semantic conflicts between their concepts and those of global ontology. These methods are based either on the degree of relationships between concepts (i.e., equivalence, sibling, etc.) or on the computation of semantic distance between concepts.

## 5 Conclusion

Nowadays, ontologies are unavoidable to achieve semantic interoperability. The proposed approach enables thus, to decrease the number of need translations between N different Workflow. Instead of implementing  $N(N-1)$  transformations (without the central OWL ontology), we only implement  $2N$  transformations. Finally, our approach is in accordance with the MDA principle based on translations between PIM (Platform Independent Model) and PSM (Platform Specific Model) for generating the code. The common meta-model and ODM are then considered as two PIMs and the OWL meta-model is the PSM.

## References

- [1] Hamri.S, Boufaida.M, Boudjlida.N, “ An Architecture for the Interoperability of Workflow Models ”, In Proceedings of the first international workshop on Interoperability of Heterogeneous Information Systems, ACM, Germany, 2005, p 31.
- [2] “ebXML Business Process Specification Schema Version 1.01”, ebXML Business Process Project Team, mai 2001, <http://www.ebXML.org/specs/ebBPSS.pdf>
- [3] WfMC, Proposal for an Asynchronous HTTP binding of Wf-XML,, June 1, 2000.
- [4] Chauvet. J.M, « Web Services avec SOAP,WSDL, UDDI, ebXML...», Eyrolles 2002.
- [5] Satish Thatte, « XLANG, Web Services for business process design », Mai 2001.
- [6] « Web Services Choreography Interface 1.0 », BEA systems, Sun Microsystems, 2002.
- [7] Workflow Management Coalition, Workflow Process Definition Interface, XML Process Definition Language, Document Number TC-1025, May 22, 2001.
- [8] Jean Bézivin, “Who’s Afraid of Ontologies”, OOPSLA’98 Workshop : Model Engineering, Methods and tools Integration with CDIF, Vancouver October 1998.
- [9] D.Djuric, D.Gasevic and V.Devedzic, “ Ontology Modeling and MDA”, Journal on Object Technology, Vol 4, N 1, January-February 2005, Online at <http://www.jot.fm>.
- [10] “OWL Web Ontology Language Reference, W3C Recommendation (2004) <http://www.w3.org/TR/2004/REC-owl-ref20040210/>,
- [11] Gašević, D. Et al : “Bridging MDA and Ontologies ”, Journal of Web Engineering, Vol, 4, No.2 (2005) 118-143.
- [12] <http://protégé.stanford.edu/plugin>.
- [13] Meta Object Facility (MOF) Specification v1.4 OMG Document format/02-04-03, (2002), <http://www.omg.org/cgi-bin/apps/doc,format/02-04-03.pdf>.