An Approach for Building an OWL Ontology for Workflow Interoperability

S. Hamri[1], N. Boudjlida[2], M. Boufaida[1]

[1]LIRE Laboratory, Mentouri University of Constantine, Algeria

[2]Loria Laboratory, Poincarré University of Nancy1, France

Summary

Introduction

- Workflow interoperability:
  - Models
  - Engines
- In the field of Workflow interoperability (or business process), several modeling languages have been proposed: XPDL, WSFL, XLANG, BPEL, WSCI, WSCL, ebXML, BPML, etc.
- However, no language has been adopted as a standard for Workflow interoperability and no common meta-model has been agreed upon.

Related Work (1)

- Several works related to Workflow interoperability have been conducted leading to multiple process modeling languages (XPDL, XLANG, WSFL, BPEL4WS, WSCI, …)
- Although these works have treated the interoperability in Workflow domain, there is **no semantics** at higher levels of abstraction.
- Indeed, they generally provide **a canonical model**, which is insufficient, such as XPDL, BPEL4WS, etc.

Related Work (2)

- ✓ *No Common standard* has been agreed upon and *No formal Semantics* for the concepts of these languages.
- ✓ Furthermore, no *common meta-model* (XPDL, PIF (Process Interchange Format), etc.) has been adopted.
- ✓ Hence, the approach that we propose, ***supports the semantic interoperability.***

5

Overview of the Approach (1)

- ❑ If we consider a Workflow as a support for business activities, it is necessary to take into account the knowledge context of these activities.
- ❑ Indeed, use of ontologies is one mean to consider this kind of knowledge.

6

Overview of the Approach (2)

- ❑ We propose an ontology-based approach for building an OWL Workflow ontology for Workflow interoperability.
- ❑ It constitutes then, a common ontology that aims at making Workflow models understand each other.
- ❑ To give meaning to the exchanged information by using a shared ontology between Workflows

7

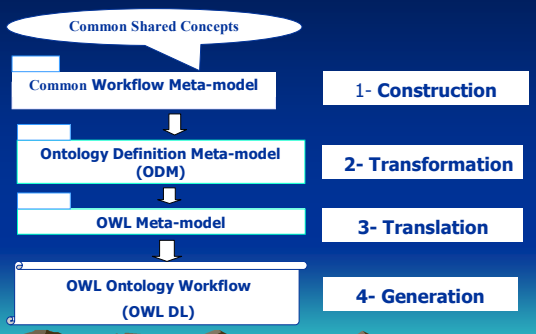
Overview of the Approach (3)

- ❑ Therefore, to use an ontology language (i.e, OWL) as a reference Language that makes Workflow users understand each other.
- ❑ Finally, to focus on an architecture that supports our approach for Workflow interoperability.
  - ✓ Therefore, the process for building this OWL ontology is defined by the following steps.

8

Overview of the Approach (4)

Steps for building an OWL ontology Workflow :



Overview of the Approach (5)

- ODM was designed to enclose ontology concepts
- OWL is the result of the evolution of existing representation languages (RDF, RDFS, etc.) and is a W3C recommendation for publishing and shared ontologies in Semantic Web.

Overview of the Approach (6)

- For the transformation steps (*step 2 and step 3*), we use the corresponding table of mappings between MOF concepts, ODM concepts and OWL concepts.
- Finally, in *the last step*, we use the ontology tool Protégé for generating the OWL definition of the ontology

Overview of the Approach (7)

Principle :

- We combine the MDA (Model Driven Architecture) approach with ontological engineering.

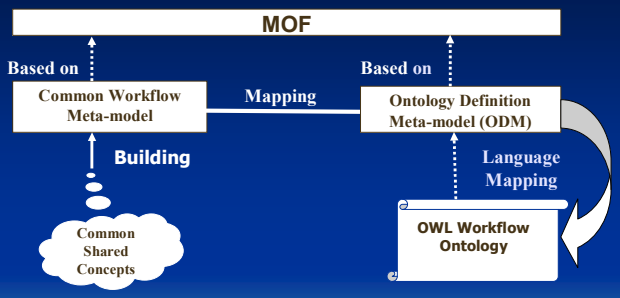
- *MDA is used :*
  1. For building a common Workflow meta-model based on MOF (Meta-Object Facility).
  2. With using an Ontology Definition Meta-model (ODM) using MOF and based on OWL (Ontology Web Language).
  3. And using an OWL meta-model based on MOF.

Overview of the Approach (8)

- *Ontological engineering :*
  - For exploiting ontologies that define and position the concepts that describe the knowledge of the Workflow domain.
  - And using OWL DL (Description Language) that provides the tool Protégé OWL Plugin for the generation of the OWL ontology.

➡ A combination for constructing an OWL Workflow ontology.

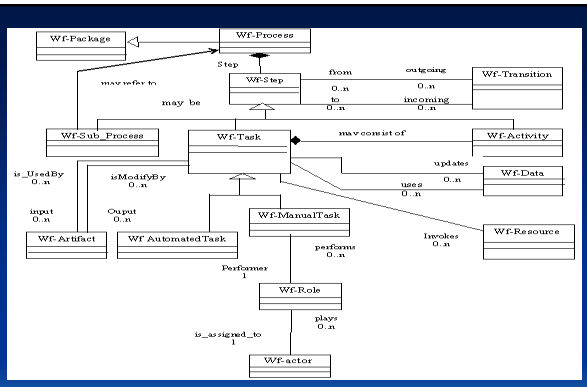
Overview of the Approach (9)



Steps for constructing an OWL Workflow ontology from common concepts.

The Common Workflow Meta-model

- To build this meta-model, *the first step* is to investigate the concepts that are common and shared between the most Workflow models.
- The extracted concepts have been compared and have been aligned up according to their objectives and the semantic definition of concepts as defined by their designers.
- This common meta-model is considered as a common ontology.



The Common Workflow Meta-model.

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

Mappings between MOF, ODM 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

Table 1. Summary of Mappings between MOF, ODM and OWL concepts.

- However, the transformation from ODM to OWL is straightforward since ODM construction is based on OWL (ODM and OWL have similar concepts).
- Example of translation of two MOF-Classes: Wf-Task and Wf-ManualTask, which is a SubClass of Wf-Task into OWL description.

MOF-Class	Owl Equivalent
Wf-Task	<owl:Class rdf ID = "Wf-Task"
Wf-ManualTask	<owl:Class rdf ID = "Wf-ManualTask" < rdfs : SubClassOf rdf : resource = " #Task"/> </ owl:Class

Conclusion (1)

- Using a single approach in solving interoperability problems is usually not enough.
- The MDA approach is insufficient for achieving semantic interoperability because its standards (MOF and XMI (XML Metadata Interchange)) do not guarantee completely the semantics of models.

Conclusion (2)

- Therefore, integrating MDA approach for the benefit ontological engineering is a good idea.
  - ✓ MDA for its advantages (portability, platform independence, etc.).
  - ✓ Ontological engineering since ontologies allows us to define and to position the concepts that describe our domain and to define their semantics.

21

Conclusion (3)

- ✓ Since OWL has an XML-based representation, we can use XSLT for the transformations from source model to target model via OWL (all meta-models (source and target) are MOF-compliant languages.
- ✓ This approach relies on an architecture (so-called an MDA-defined ontology architecture) whose advantages are openness, flexibility and evolution.

22

Conclusion (4)

- ***Advantages of the approach :***
  - ❖ The common meta-model is generic and re-usable. So, it may be used in different business process contexts.
  - ❖ Building an OWL ontology via an Ontology Definition Meta-model (ODM) is open and flexible since when one wants to support a new language (e.g., DAML+OIL), she/he only uses the ODM-based principle.

23

Conclusion (5)

- ❖ Furthemore, the proposed approach enables thus, to decrease the number of need translations between N different Workflow models (2*N transformations instead of N*(N-1) transformations).
- ❖ Finally, the approach is in accordance with the MDA principle based on translations between PIM (Platform Independent Model) and PSM (Platform Specific Model): the common meta-model and ODM are then considered as two PIMs and the OWL meta-model plays the role of the PSM.

24


THANK YOU
FOR YOUR
ATTENTION ……
25