OWL Change Management Patterns

Rim Djedidi[1], and Marie-Aude Aufaure[2]

[1] Computer Science Department, Supélec Campus de Gif
Plateau du Moulon – 3, rue Joliot Curie – 91192 Gif sur Yvette Cedex, France
rim.djedidi@supelec.fr

[2] MAS Laboratory, SAP Business Object Chair –Centrale Paris
Grande Voie des Vignes, F-92295 Châtenay-Malabry Cedex, France
marie-aude.aufaure@ecp.fr

Abstract. Ontology evolution is a complex problem. In our work, we focus on issues related to change management, particularly consistency maintenance and we present in this paper, an overview of Change Management Patterns (CMP) that we have defined to model the three dimensions *change, inconsistency* and *resolution alternative*. Modelling these patterns and the links between them, helps to propose an automated process guiding and controlling change application while maintaining consistency of the evolved ontology.

Change management depends closely on the ontology representation model, we focus on OWL language and we consider change impact on logical consistency as specified in OWL DL layer.

Keywords: Ontology Evolution, Change Management, Consistency, OWL DL.

1 Introduction

In this paper, we present an overview of OWL Change Management Patterns (CMP) that we have defined. The goal of pattern-centered modelling is to offer for each of the three dimensions: *change*, *inconsistency* and *resolution alternative*, different levels of abstraction and to establish links between them (Fig 1).

2 Change Management Patterns

Change Management Patterns (CMP) are proposed as a solution looking for invariances in change management that repeatedly appear when evolving ontologies. They classify categories of changes based on OWL model, categories of logical inconsistencies considering OWL DL constraints and categories of inconsistency resolution alternatives. Links between CMP help determining inconsistencies that could be potentially caused by a type of change and alternatives that possibly resolve a kind of inconsistency and thus providing an automated process for change management.

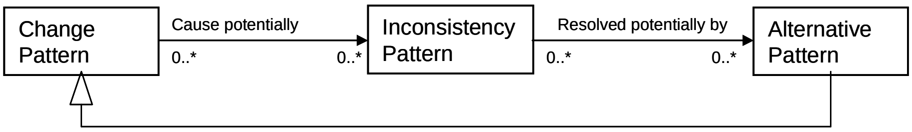

Fig. 1. Conceptual model of change management patterns (CMP).

2.1 Change Patterns

Change patterns categorize changes and define formally their signification, their scope and their potential implications. They cover OWL basic changes and a first core of composed changes. Change pattern components are:

- Involved entities corresponding to conceptual primitives of OWL;
- Arguments including all necessary parameters to implement the change. Argument content varies depending on change type and involved entities;
- Constraints (preconditions) to satisfy so that change can be applied without affecting the logical consistency of the ontology.

Example1. Let’s consider a basic change pattern of adding a sub-class:

Table 1. Basic change pattern example.

Type	Involved Entities	Arguments	Constraints	OWL DL Axioms
P_Bas_Chg_ Add_ Sub_Class	Class, Class	Id_Super-class Id_Sub-class	`¬(sub_classID disjointWith Super_classID)`	`SubClassOf (Super_classID ,Sub_classID)`

2.2 Inconsistency Patterns

Inconsistency pattern components are:

- The IDs of all entities implicated directly or indirectly in a logical inconsistency;
- The IDs of entities involved in the inconsistency;
- Axioms involved in the inconsistency.

Example 2. Let’s take the basic change example (Example 1) and suppose that the constraint is not satisfied, a possible corresponding disjointness inconsistency pattern can be described as follow:

Table 2. Inconsistency pattern example.

Type	Implicated Entities	Involved Entities	OWL DL Axioms

P_Incons
_Disj
	Super_classID,
	Sub_classID,
	Super_sub_class_ID
		Super_classID,
		Super_sub_
		class_ID
			`Super_sub_class_ID ⊑ ¬Super_classID,`
			`Sub_classID ⊑ Super_sub_class_ID`

2.3 Alternative Patterns

An alternative pattern represents an additional change to apply so that a logical inconsistency can be resolved. It is described as a basic or composed change and it inherits change pattern properties (Fig 1). Other information can also enrich alternative pattern description such as preconditions to satisfy before choosing an alternative pattern as a resolution.

Example 3. Let's reconsider the basic change example (Example 1), the possible alternative patterns for disjointness inconsistency caused by this change (Example 2) can be described as follow:

Table 3. Alternative pattern examples.

Pattern alternative 1	
Type	P_Alt_Disj_ Bas_Chg_Add_ Sub_Class
Involved Entities	Class, Class
Arguments	Id_Superclass, Id_Sub-class, Id1_cls_disj, Id2_cls_disj
Preconditions	SuperClass(Id1_cls_disj)∩ SuperClass(Id2_cls_disj) = Id_Superclass
Constraints	¬(sub_classID disjointWith Super_classID)
OWL DL Axiom	SubClassOf (Super_classID, Sub_classID)
Pattern alternative 2 (synthetic version)	
Type	P_Alt_Disj_Comp_Chg_HybridClass_Link
Involved Entities	Class, Class
Arguments	Id_ HybridClass, Id_Sub-class, Id1_cls_disj, Id2_cls_disj
Intermediate Components : OWL DL Axioms	Class(Id_ HybridClass {intersectionOf({ Id1_cls_disj, Id2cls_disj}) SubClassOf (Id_ HybridClass, Id_sub_class)

3 Conclusion

In this paper we present OWL Change Management Patterns modelling the three dimensions: *change*, *inconsistency* and *resolution alternative* and links between them. The defined patterns cover OWL DL basic changes, a sub-set of composed changes and a first core of logical inconsistencies and of alternatives that could resolve them.

Acknowledgments. Researches reported in this paper are founded by the French National Research Agency ANR, as a part of the project DAFOE: Differential and Formal Ontology Editor.