Dynamic Ontology Evolution in Open Environments

Ignazio Palmisano[1], Valentina Tamma[1], Luigi Iannone[2], Terry Payne[1] Paul Doran[1]
{ignazio, V.Tamma, luigi, trp, P.Doran}@liverpool.ac.uk

[1] Department of Computer Science, University of Liverpool
[2] Department of Computer Science, University of Manchester

Abstract. Changes in an ontology may have a disruptive impact on any system using it. This impact may depend on structural changes such as name changes or relations between concepts, or it may be related to a change in the expected performance of the reasoning tasks. As the number of systems using ontologies is expected to increase this problem is likely to occur more frequently, and, given the open nature of the Semantic Web, new ontologies and modifications to existing ones are to be expected. Dynamically handling these changes, without requiring human intervention, is a key requirement for successful applications. This paper presents a system that isolates groups of related axioms in an OWL ontology, so that a change in one or more axioms can be automatically localised to a part of the ontology. In addition, we report the results on evaluating the effectiveness of our approach on large ontologies.

1 Introduction

Interoperation between knowledge-based systems or agents[3] requires a common vocabulary to facilitate successful information exchange, and it is now accepted that this common vocabulary is represented within *ontologies* [1]. However, with the recent evolution and ubiquity of networks, and in particular of the World Wide Web, documents, systems and services are becoming increasingly decoupled, distributed, and decentralised [2], and more significantly, are thus situated in *open environments*. In this context, we characterise open environments as enviromnets where documents, systems and services may appear, change or dissapear at any time, and thus no assumptions can be made about content, interaction protocol, or even availability or existence! Whilst the inherent decentralisation and scale of the World Wide Web has facilitated a revolution in the way information is now disseminated (globally), the notion of there being common domain ontologies sufficient for catering to the requirements of a diverse range of consumers and produces of services has become untenable. Likewise, given this open

[3] The term agent is here used loosely, to include intelligent agents, but also any computer system capable of autonomously performing a task.

characteristic, and the scale of the web, these heterogeneous entities are becoming autonomous, necessitating the need for dynamic system configuration [2] and intelegent adaptaion. Ontology alignment provides a pragmatic (and principled) mechanism for facilitating interoperation between different ontologies, and thus interaction between agents and services.

Determining alignments, however, is a computationally expensive process that is traditionally performed offline. Whilst the use of such alignemnts can be readily used when services are known a-priori, the use of alignments at run-time within an open environment requires both mechanisms for locating existing alignments, and then selecting one (from a set of several possibilities) to facili-ate service communication [5]. This process can be conputationally heavy, and potentially redundant if a community of agents transact on a regular basis over a period of time. A small number of approaches have addressed how such cases can be managed optimally, by considering: the notion of evolving ontologies dynamically [4]; dynamic determination of alignments [5]; or the emergence of semantics [6]. In these types of environments it is necessary to consider that no ontology can be expected to remain unchanged throughout its lifetime, but will respond to changes in the environment, such as a change in the data represented in the ontologies, the need to accommodate the arrival of new agents, the improvement of the efficiency of repeated communication between a group of agents, or the dynamic determination of the alignment of their ontologies.

Traditional ontology evolution approaches (for instance the one in [7]) are semi-automated at best, and assume manual guidence from one (or several) domain experts. However, they rarely consider the use of estimates on the impact resulting from changes to an axiom on the whole ontology. Estimating this effect *a priori*, i.e. before performing the change itself, is even more crucial in open environmnets, where dynamism and large scale prevent any human intervention [8]. In this case, the agents' ability to acquire new capabilites and therefore to achieve new tasks (or answer new queries, in case of knowledge based systems) needs to be offset by the *cost* of the change in terms of employment of scarce resources [8], and with partial knowledge of the environment, typical of agents [9]. Some efforts have demonstrated that the addition of new axioms to the agent knowledge base increases its ability to achieve a task [10] but only few efforts have attempted to estimate the impact of change, and not for dynamic evolution [11]. When the evolution concerns agent ontologies, the ability to estimate the change can be used to deliberate on the usefulness of change in a *rational* fashion on the grounds of the complexity of the modifications required.

In this paper we present an approach that evaluates the impact of change on an ontology a priori, without using reasoning, but by using a crude estimate of the set of axioms in an ontology that are impacted by the change. This work is inspired by the notion of *bounded rationality* [12] because it assumes that the agent's decision making process is optimised to work with partial knowledge, and with limited computational resources. The approach determines the maximal scope of the effect of the modification performed in an ontology (*group*). The idea is that by identifying a group, agents can determine the impact of a proposed change, identify the reclassification costs involved, and therefore decide *a priori*, and without having to make use of reasoning, whether performing the change to their ontology is in their best interest. This work concentrates on introducing the formal definition of a group (Section 3), and the discussion of the properties that it exhibits (Section 4). Section 5 presents an empirical evaluation of the system, and the paper concludes by presenting final remarks and future directions.

2 Ontology Modification Impact

Reasoning complexity in Description Logics depends on the selection of constructors and in the way they are combined. Whilst reasoning on some less expressive variants of Description Logics has been formally proved to be tractable in worst case scenarios, many expressive Description Logics subsets (which differ for the set of available constructors) exhibit exponential worst case complexity, e.g., in $\mathcal{SHOIN}(\mathcal{D})$ the concept satisfiability test is NExpTime-complete. However, common use does not always push the complexity to the limit, therefore, it has become practically possible to use DL based ontologies in real systems, as detailed in [13].

In the presented approach, in order to partially evaluate the impact of an axiom over the complexity of reasoning for a real world ontology (considering both T-Box and A-Box, i.e. the set of axioms about classes and properties and the set of assertions about individuals), each axiom and its interactions with the rest of the ontology are considered, aiming to build a knowledge base in which the impact of changes can be evaluated in terms of the expected computational load that they represent.

The analysis that can be carried out is necessarily approximate, since the aim is to evaluate the cost of a change *before* the change takes place, and a complete evaluation is only possible *after* the change has taken place. Therefore, the impact of a change is measured using the very simple heuristic of the number of axioms and assertions contained in the section of knowledge base affected by the change.

W.r.t. the alignment example, the modification consists in the addition of Ax and Def:

$$Ax = C_2 \ \subseteq \ C_1; \quad Def = C_2 \ \subseteq \ \neg(C_4 \ \sqcap \ C_3)$$

to a knowledge base K:

$$K = \{C_1 \subseteq SC_1,\ C_1 \subseteq SC_2,\ SC_2 \subseteq SC_4\}$$

K has expressivity $\mathcal{AL}$; adding Ax and Def, as represented in Figure 1, would change the expressivity of the knowledge base to $\mathcal{ALC}$, changing then the computational complexity in the worst case.

Worst case complexity, however, is not always the complexity the reasoner actually has to face; in order to evaluate more precisely the impact of change, the parts that will be affected by the change will be selected.

When adding an axiom that would raise the expressivity of the ontology, then, this increase will be restricted to a portion of the knowledge base. In this way, an estimate based on the expressivity of the knowledge base and on the number of axioms involved will yield a more precise measurement of the computational effort required.

Given the need for a computationally light way to assess the impact, the grouping framework is designed to sacrifice minimality of the impacted parts; for the same reason, the metric used is very simple to compute, in order to cope with very limited resources. Details on how the number of the impacted axioms might be reduced, at the expense of the required effort, are given in Section 6.

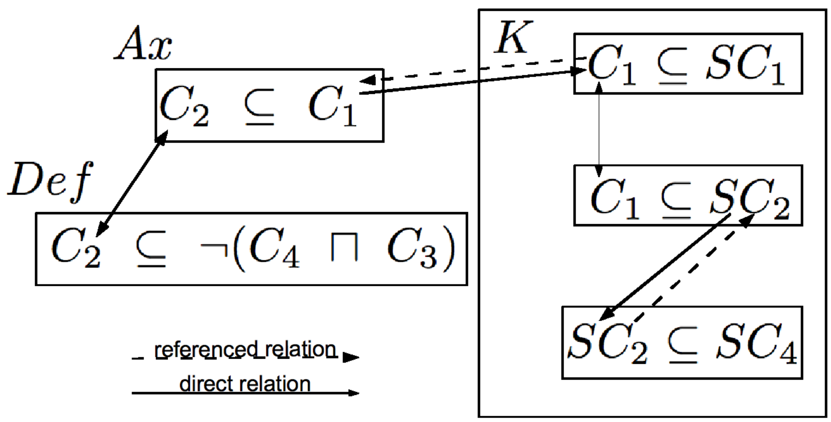

Fig. 1. K Knowledge base: the group rooted at C_1 contains the whole knowledge base

3 Framework Definition and Implementation

Let us define an *axiom* A representing Ax in the framework:

- an *axiom* A is the basic unit to build ontologies; it represents the abstraction over an OWL axiom or assertion, internally represented as a set of RDF[4] statements[5], which has a *concept signature* and a *predicate signature*, and is represented as a node in a directed multigraph whose edges represent *relations* between *axioms*;
- a *concept signature* cs is the set of named and unnamed concepts appearing in A; in the example, A has $cs = \{C_1, C_2\}$;

[4] `http://www.w3.org/RDF`

[5] as in OWL Web Ontology Language Semantics and Abstract Syntax Section 4. Mapping to RDF Graphs `http://www.w3.org/TR/owl-semantics/mapping.html`

– a *predicate signature* ps is the set of roles mentioned in A; in the example, the ps of A is empty;
– a *main concept* or *main role* m, which is the concept or role being defined by the *axiom*[6]; in the example, m for A is C_2;
– a *signature* $S = S = cs \sqcup ps \sqcup \{m\}$ is the union of *concept signature*, *predicate signature* and *main concept* or *main role*;
– *relations* between *axioms* amount to intersections between their *signatures*; they can be:
  - *direct*: a *direct* relation between an *axiom* A and an *axiom* B exists if m for B belongs to the cs or ps for A; two *axioms* that share m have a bidirectional *direct* relation; this is the case with Ax and Def, for which m is C_2;
  - *indirect*: an *indirect* relation between two *axioms* A and B holds when S for A overlaps with S for B through a node different from m of either A or B (e.g. $\exists\ R.C$ and $\exists\ R.D$ share a reference to the role R in their ps); an *indirect relation* is bidirectional;
  - *referenced*: a *referenced* relation is the inverse of a *direct* relation; such a relation is implicitly defined also for *indirect* relations, in which case it is bidirectional as the *indirect relation*.

In DL, a signature is defined as the disjoint union $\mathbf{S} = \mathbf{R} \uplus \mathbf{C} \uplus \mathbf{I}$ of sets of *role names* (denoted by R, S,...), *concept names* (denoted by A, B,...) and *nominals* (denoted by i, j, k...), appearing in an ontology; however, the definition given above to describe single axioms includes anonymous concepts, that in DL terms are only syntactic structures. This reflects more closely the abstract syntax for the OWL language[7] and simplifies the implementation, enabling a complete abstraction from the underlying RDF translation. To this extent, a concept in a *concept signature* corresponds to a URI or an unnamed (blank) node (i.e. RDF nodes without a label) representing an OWL class, while a predicate in a *predicate signature* corresponds to a URI being used as a predicate in the underlying RDF representation; URIs in the language namespaces, i.e. OWL, RDF, RDFS[8] and XML Schema Datatype[9], are ignored when computing signatures.

The three kinds of relations define three graphs defined as follows:

– O is the set of *axioms* in an ontology $\mathcal{O}$;
– O_{direct} is the graph $< O, DirectRelations >$ where *DirectRelations* is the set of edges that represent *direct* relations;
– $O_{indirect}$ is the graph $< O, IndirectRelations >$ where *IndirectRelations* is the set of edges that represent *indirect* relations;

[6] m is referred to as *main node* at `http://www.w3.org/TR/owl-semantics/mapping.html`. It is not defined for all OWL axioms; for those cases in which it is left undefined, m is assumed to be the common subject of all the triples involved in the *axiom*.
[7] `http://www.w3.org/TR/owl-semantics/direct.html`
[8] `http://www.w3.org/TR/rdf-schema/`
[9] `http://www.w3.org/XML/Schema`

- $O_{referenced}$ is the graph $< O, ReferencedRelations >$ where $ReferencedRelations$ is the set of edges that represent the *referenced* relations.

Definition 1 (Group rooted at an axiom). *A group G rooted at an axiom A is the set of axioms resulting from the union of the sets of axioms S_{direct}, $S_{indirect}$ and $S_{referenced}$ explored during the exhaustive visit of O_{direct}, $O_{indirect}$ and $O_{referenced}$ respectively, starting from A and following the relations.*

Depending on the ontology, the size of a group can vary from a few axioms to the whole ontology; some results on real ontologies are reported in Section 5.

Let us take into consideration the two simplest modifications to the knowledge base: the introduction of a new *axiom* X or the removal of an *axiom* Y already in the knowledge base.

In the first case, X may have relations to one or more *axioms*, and therefore to one or more groups:

- X defines a new group, rooted at X, not influencing the existing groups; this is the case in which X has only *direct* outgoing relations to other axioms; the change then only impacts the new group;
- X may be part of an existing group or act as a bridge between different groups; this happens when X has *direct* or *indirect* incoming relations from one or more group; the change impacts all involved groups.

In the second case, the removal of Y can have the following effects:

- Y may have relations to only one group and its removal does not break the group into subgroups; in this case, only the group rooted at Y is impacted;
- Y may have been acting as a bridge between two or more groups; those groups will then be separate, and this will require reclassification on the new groups.

Modifications to one or more axioms can be reduced to a composition of these two kind of changes: the modification of a single *axiom* is equivalent to the removal of the previous form of the *axiom* and to the introduction of the new form, while, due to DL monotonicity, the order in which multiple modifications are made is not relevant to the final result.

Given a set of changes, the framework is able to select the affected groups and therefore reduces the size of the ontology that needs to be submitted to the reasoner; on the other hand, the groups of independent *axioms* are not disjoint (grouping the *axioms* is not a partition in the mathematical sense), therefore there is a certain amount of duplication in the reasoning; this issue is discussed further in Section 5, together with an analysis of the overlap between different groups.

The advantage of the framework is that determining the groups does not require a DL reasoner; it is in fact sufficient to take into account the syntactic characteristics of an ontology. The clear advantage is that the heavy computational load of reasoning is not required until the knowledge base is actually used, thus enabling the use of grouping over large ontologies. On the other hand, the signature overlap between *axioms* is not a sure sign that the *axioms* are related, but it is only an indication; therefore, groups tend to be larger than strictly necessary, since not all relations mean that the related *axioms* are used to entail relevant knowledge.

The grouping framework is implemented in Java; it uses Jena [14] and the SPARQL [15] implementation ARQ[10] to perform the *axiom* extraction. In order to carry out reasoning tasks over groups, the OWL DL reasoning engine Pellet [16] has been used, in order to check the expressivity of a group, as reported in Section 5, and to verify whether reasoning over each group and collecting all inferred axioms produces the same set of inferred axioms as reasoning on the whole ontology.

4 Theoretical Properties of Groups

As defined in Section 3, a group consists of a set of related *axioms*. DL monotonicity guarantees that reasoning on a group G of a knowledge base K is *correct*, in the sense that no *axiom* that can be inferred from G cannot be inferred from K. This is referred to as *local correctness* in [17], where the concept of *uniform interpolant* [18, 19] is applied to prove the properties of a module extraction technique.

Local completeness would be a desirable property for a group, i.e., any DL axiom that can be proved w.r.t. K and which depends on DL axioms in G should be provable w.r.t. G alone. However, to the best of our knowledge this is still an open problem, since it depends on the ability to prove that G is a uniform interpolant for K. While a formal proof of this property for groups is not available, other relevant considerations may be done.

Let us consider the $\mathcal{ALC}$ DL and the corresponding tableaux expansion rules, as defined in [13], then, given an *axiom* A, all *relevant* entailments that depend on A only depend on *axioms* contained in G, the group rooted at A.

To support this claim, let us consider the expansion rules:

(1) $\sqcap$**-rule**: if $\mathcal{A}$ contains $(C_1 \sqcap C_2)(x)$ and does not contain $C_1(x)$ and $C_2(x)$, then $\mathcal{A}_1 = \mathcal{A} \bigcup \{C_1(x), C_2(x)\}$.
(2) $\sqcup$**-rule**: if $\mathcal{A}$ contains $(C_1 \sqcup C_2)(x)$ but neither $C_1(x)$ nor $C_2(x)$, then $\mathcal{A}_1 = \mathcal{A} \bigcup \{C_1(x)\}$, $\mathcal{A}_2 = \mathcal{A} \bigcup \{C_2(x)\}$.
(3) $\exists$**-rule**: if $\mathcal{A}$ contains $(\exists R.C)(x)$, but there is no individual name z such that $C(z)$ and $R(x, z)$ are in $\mathcal{A}$, then $\mathcal{A}_1 = \mathcal{A} \bigcup \{C(y), R(x, y)\}$ where y is an individual name not occurring in $\mathcal{A}$.
(4) $\forall$**-rule**: if $\mathcal{A}$ contains $(\forall R.C)(x)$ and $R(x, y)$, but it does not contain $C(y)$, then $\mathcal{A}_1 = \mathcal{A} \bigcup \{C(y)\}$.

Let us consider rule (1): the possible situations are:

[10] http://jena.sourceforge.net/ARQ/

a $\mathcal{A}$ contains the *axiom* $A{=}X \equiv C_1 \sqcap C_2$, and the assertion $X(c)$ (also represented as an *axiom* in the framework); these two are both contained in G, the group rooted at A, by means of a *direct* relation over X;

b $\mathcal{A}$ contains an assertion $C_1(y)$, which is relevant for the application of the $\sqcap$-rule; this assertion is linked to A through an *indirect* relation and therefore is included in G;

Any axiom or assertion relevant for rule (1) w.r.t. A will be contained in G; therefore, restricting the input on which to apply the $\sqcap$-rule to G does not preclude any entailment in $\mathcal{A}$.

The same approach works with rule (2): any assertion in $\mathcal{A}$ relevant for rule (2) will be included in G rooted at $A{=}X \equiv C_1 \sqcup C_2$.

For rules (3) and (4), the relation between the *axioms* ($X \equiv \forall R.C$ and $X \equiv \exists R.C$) and the role assertions ($R(x,z)$ and $R(x,y)$) is an *indirect* relation based on the occurrence of R in both; The result in both cases is that all the relevant axioms and assertions are included in the groups G_1 rooted at $A_1{=}X \equiv \exists R.C$ and G_2 rooted at $A_2{=}X \equiv \forall R.C$ respectively.

Therefore, it seems reasonable to think that most of the relevant inferences that are entailed by $\mathcal{O}$ are also entailed by at least one group G computed over $\mathcal{O}$. Empirically, this has been proved true for the ontologies used in the evaluation, except for full Galen, where the test is harder to perform due to the size and number of groups; these considerations have not yet been extended to cover the higher expressivity of these ontologies (e.g. all of them include functional roles, which are not covered in the above).

4.1 Counterexamples for Local Completeness

There are two main counterexamples for the *local completeness* of a group:

(a) Any unsatisfiable axiom, such as $\top \subseteq \bot$, will affect the whole ontology.

This may be seen as a false problem: an ontology containing a contradiction may be used to entail anything, therefore its use in a real system is dubious; the grouping framework is agnostic w.r.t satisfiability in the input ontology, since no reasoning is used to compute the groups, but it is reasonable to assume that only satisfiable ontologies would be used in order to draw useful conclusions; countermeasures to enable a system to reason with inconsistent ontologies [11, 20] are outside the current scope of the framework[11].

(b) It is possible to define a concept C that can be deduced to be $C \equiv \top$; C therefore includes any concept defined in any group, but not all groups would include C. This is founded on the following definition:

$$K = \{C \subseteq \top, C_1 \subseteq C, C_2 \subseteq C, C_1 \equiv \neg C_2\}$$

[11] It is worth noting that this specific counterexample might be fetched as input to the current implementation, by adding $\top \subseteq \bot$ to any ontology; the result would be that none of the computed groups would contain this axiom, since it is only composed of language concepts and roles; therefore, grouping would actually fix the ontology by splitting it into satisfiable fragments.

In K, C would be deduced equivalent to $\top$, because of C_1 being the complement of C_2 w.r.t. $\top$, and $C_1 \sqcup C_2 \subseteq C$; but $C_1 \sqcup C_2 \equiv \top$, hence $C \equiv \top$. A generic group G, however, would not include C and its subclasses unless some other axiom in K refers to C, C_1 or C_2, and therefore the axioms of the form $C_i \subseteq C$ for each C_i mentioned in K would not hold w.r.t. the majority of groups. While this is a possible loss of information, on the other hand the construct that generates this effect is questionable, since in many cases it would be considered a modelling error in the input ontology; therefore, missing these deductions should not hamper the performance of most real world systems.

5 Practical Evaluation

In order to test the effectiveness of axiom grouping, the groups that can be isolated in real world ontologies have been investigated.

The most interesting ontologies used are:

- The Galen ontology translated in OWL[12],
- A fragment of Galen[13], smaller than the original and often used to test reasoners,
- Fisheries commodities[14] and Land Areas[15], used in the NeOn project[16] in the context of its FAO[17] case study,
- the Adult Mouse Anatomy[18], included in the Ontology Alignment Evaluation Initiative (2007 Contest)[19].

Ontology	Number of Axioms	Number of Groups	Average Axioms per Group
Galen full	82030	37408	9649
Galen fragment	9915	4500	8529.32
Commodities	1423	1422	1419.97
Mouse Anatomy	5431	3455	2925.37
Land	338	327	333.95

Table 1. Preliminary Experimental Results: Size Metrics

The measures taken into account are: the number of axioms in the ontology, the average number of axioms in a group and the expressivity of each group (Table 1); for each expressivity level, the number of groups (duplicate groups are counted as one) and the size range are reported in Table 2.

[12] `http://www.co-ode.org/galen/`
[13] `http://www.daml.org/ontologies/400`
[14] `http://www.fao.org/aims/aos/fi/commodities_v1.0.owl`
[15] `http://www.fao.org/aims/aos/fi/land_v1.0.owl`
[16] `http://www.fao.org/aims/neon.jsp`
[17] Food and Agriculture Organization of the United Nations, `http://www.fao.org`.
[18] `http://webrum.uni-mannheim.de/math/lski/align2007/mouse_anatomy.owl`
[19] `http://oaei.ontologymatching.org/2007/`

Ontology	Expressivity	Groups	Size range
Galen full	$\mathcal{AL}$	72	1/45
	$\mathcal{ALC}$	14303	61723/63625
	$\mathcal{ALCHIF}$	28	13/27
	$\mathcal{ALR}+\mathcal{HI}$	1	1
	$\mathcal{ALR}+\mathcal{HIF}$	313	53/1739
	$\mathcal{SHIF}$	24711	93/81815
Commod-ities	$\mathcal{ALCHF}(\mathcal{D})$	7	1407/1422
	$\mathcal{ALHF}(\mathcal{D})$	2	1376/1380
Mouse Anatomy	$\mathcal{AL}$	378	1/345
	$\mathcal{ALC}$	2374	3/5070
Land	$\mathcal{ALCF}(\mathcal{D})$	2	277/280
	$\mathcal{ALCHF}(\mathcal{D})$	13	299/336
	$\mathcal{ALHF}(\mathcal{D})$	10	259/269

Ontology	Expressivity	Groups	Size range
Galen fragment	$\mathcal{AL}$	552	1/115
	$\mathcal{AL}(\mathcal{D})$	441	3/255
	$\mathcal{ALC}$	101	3/151
	$\mathcal{ALC}(\mathcal{D})$	1257	4/7576
	$\mathcal{ALCH}$	1	14
	$\mathcal{ALCH}(\mathcal{D})$	29	20/7370
	$\mathcal{ALCHF}$	3	4/45
	$\mathcal{ALCHF}(\mathcal{D})$	95	16/7904
	$\mathcal{ALH}$	133	2/103
	$\mathcal{ALH}(\mathcal{D})$	7	4/79
	$\mathcal{ALHF}$	52	2/98
	$\mathcal{ALHF}(\mathcal{D})$	16	4/508
	$\mathcal{ALR}+\mathcal{H}$	7	2/29
	$\mathcal{ALR}+\mathcal{H}(\mathcal{D})$	1	25
	$\mathcal{ALR}+\mathcal{HF}$	29	5/170
	$\mathcal{ALR}+\mathcal{HF}(\mathcal{D})$	5	64/115
	$\mathcal{SH}(\mathcal{D})$	2	10/83
	$\mathcal{SHF}(\mathcal{D})$	427	47/9889

Table 2. Expressivity Metrics

5.1 Impact Estimation

Table 3 presents in more detail the results obtained on the Galen fragment ontology. Out of 9915 *axioms*, more than 4500 groups were computed, by selecting only *axioms* with named concepts as m. The groups overlap very often, due to the high detail of the ontology, which is rich in definitions and has a deep role hierarchy. In fact, the number of distinct groups, i.e. groups not included in any other group, is only 7. One of these group is composed of a single axiom: $Ontology(galen)$; this represents the ontology itself, and does not interact with any axiom inside the ontology. The remaining 6 groups range in size from 23 axioms to 9889, and all of them but the smallest one have a common subset containing 7469 axioms; in the following, Int will be the intersection of the 5 largest groups, labelled G_2 to G_6, and $G_i \setminus Int$ will denote the set difference between one of these groups and Int. Only two groups are significantly larger than Int, G_5 and G_6, by 1923 and 2420 axioms respectively. In Table 3, the leftmost table represents the size of the pairwise intersections between the groups, with its first column reporting the size (in number of axioms) of each of these groups; for readability, intersections of a set with itself are not reported, and neither are symmetric results.

The rightmost table reports the size of Int; for each set, then, the size of the difference $G_i \setminus Int$ and its expressivity is reported.

The expressivity of the groups is $\mathcal{ALC}$ for the four smaller groups, and $\mathcal{SHF}$ for the two larger ones; the difference in expressivity is therefore related to roles.

Given a modification to the ontology, now, it is possible to determine the part of the ontology that will be affected. Supposing the modification is the removal of an *axiom* A, it may be located at the intersection of the five largest groups, or it may be located within one specific group; using the following simple formula, computed considering the distinct groups containing A, a first approximation of the impact function may be computed:

	Size	G_3	G_4	G_5	G_6
G_1	23			6	23
G_2	7501	7485	7485	7497	7481
G_3	7488		7485	7487	7471
G_4	7503			7501	7485
G_5	9392				9376
G_6	9889				

	Number of *axioms*	Expressivity
Int	7469	$\mathcal{ALC}$
$G_2 \setminus Int$	32	$\mathcal{AL}$
$G_3 \setminus Int$	19	$\mathcal{AL}$
$G_4 \setminus Int$	34	$\mathcal{AL}$
$G_5 \setminus Int$	1923	$\mathcal{SHF}$
$G_6 \setminus Int$	2420	$\mathcal{SHF}$

Table 3. Overlaps in Galen fragment grouping

$$I(A) = \sum_{A \in G} size(G) * expressivity(G)$$

where the definitions of *size* and *expressivity* functions are as follows:

– $size()$: Given (G) the set of all groups G over an ontology $\mathcal{O}$,

$$size(G) : \mathcal{G} \rightarrow \mathbb{N}$$

is the number of *axioms* contained in G.

– $expressivity()$: Given (G) the set of all groups G over an ontology $\mathcal{O}$,

$$expressivity(G) : \mathcal{G} \rightarrow \mathbb{R}$$

is the function that computes ec, the DL expressivity of G, and maps ec into a real numeric value.

The implementation of $expressivity()$ used to compute the results presented in this paper uses a list of four expressivity classes: $EC = (\mathcal{ALC}, \mathcal{SHIF}, \mathcal{SHOIN}, \mathcal{SROIQ})$ and a list of associated weights $ECWeights$=(0.25, 0.5, 0.75, 1); the DL expressivity ec of a group G is approximated to the least expressive $\mathcal{E}$ in EC that includes ec and the weight associated with $\mathcal{E}$ is returned.

The *expressivity* function represents the weight that the reasoning has in evaluating the impact. Since worst case complexity can change with every allowed constructor in a DL, it is not simple to assign it a meaningful numeric value; as a first approximation, some values have been chosen, as reported in Table 4. Defining better approximations is one of the future developments of this work.

DL Expressivity	Complexity	Weight	OWL sublanguage
$\mathcal{ALC}$	PSpace-complete	0.25	
$\mathcal{SHIF}$	ExpTime-complete	0.5	OWL Lite
$\mathcal{SHOIN}$	NExpTime-complete	0.75	OWL DL
$\mathcal{SROIQ}$	NExpTime-hard, Decidable	1	OWL 2

Table 4. Expressivity classes weight

The values for $I(A)$ are reported in Table 5; they refer to all possible cases, i.e. A belonging to Int or to one of the other groups; on the basis of the comparison between the possible impacts, the decision making process may choose to accept the change involving A only in the cases in which the impact is smaller, which means the agent will accept the changes to $G_2 \setminus Int$, $G_3 \setminus Int$ and $G_4 \setminus Int$, therefore outside Int, while it will refuse changes to Int or to G_5 and G_6.

Observing the structure that the groups outline, a simple optimization for the reasoning task can be devised: the intersection of the groups may be considered as a common base to reason over any group, so that it is possible to reason once on the intersection and carry out the reasoning on the remaining parts of each group leveraging what already inferred. This technique would rely on the availability of incremental reasoning engines [21].

Supposing such a technique is available and Int is the intersection of all the groups, the formula for $I(A)$ can be modified as follows:
$if A \notin Int$

$$I_{incr}(A) = \sum_{A \in G} size(G \setminus Int) * expressivity(G \setminus Int)$$

$if A \in Int$

$$I_{incr}(A) = size(Int) * expressivity(Int) + \sum_{A \in G} size(G \setminus Int) * expressivity(G \setminus Int)$$

The new values for impact computed with this formula are reported in Table 5 in the column labelled $I_{incr}(A)$.

	$I(A)$ impact value	$I_{incr}(A) impactvalue$
$A \in Int$	47089.25	4060.00
$A \notin Int$: $A \in G_2$	6086.75	8.00
$A \in G_3$	6081.75	8.50
$A \in G_4$	6088.75	4.75
$A \in G_5$	14252.00	961.50
$A \in G_6$	14580.00	1210.00

Table 5. Results for impact computation

Comparing the results with those obtained for $I(A)$, it is apparent that there is an advantage in building a knowledge base around an incremental reasoning engine; such enhancement would enable the agent to be more flexible in accepting changes, since the cost of updating the knowledge base is not only known but also largely smaller than in the previous case, where a monolithic knowledge base would make the update process computationally expensive. Preliminary experiments with Pellet show that the relevant difference in the impact value between the optimised ($I_{incr}(A)$) and unoptimised ($I(A)$) approach is reflected in practice in very quick classification times for modifications outside the intersection of the groups; however, this optimised architecture is not completed at the time of writing, and therefore these results will not be presented in this paper.

6 Related Work and Possible Extensions

In [10], the authors propose a cooperation based approach in which agents exchange service descriptions; these service descriptions are not static, but may be generated at runtime by one agent and be passed to its "neighbors" so that, over time, the new service descriptions are common knowledge for the agents. Each agent needs not incorporate the new service descriptions, but it may decide to do so on the basis of some internal decision making process. The authors show, through simulations, that over time an agent tends to reuse services described by other agents instead of creating new descriptions (which is an expensive task), i.e. including new knowledge helps the agent in reaching its goals; also, including new knowledge helps the agent because it will be able to spread the acquired knowledge in its environment; from a cooperative point of view, this means that the agent will also receive the knowledge it needs from the environment.

W.r.t. the work presented in this paper, Sensoy and Yolum attack a slightly different topic: their work motivates the diffusion of knowledge between agents, proving that entities with heterogeneous knowledge will benefit from being able to exchange information and enrich their knowledge bases, but the criteria an agent should use in deciding whether to incorporate or not a specific service description are not defined; on the other hand, the main focus of this work is on devising the infrastructure that enables an agent to manage its knowledge base, tackling the growth in size and complexity with the ability to compare, e.g., two different service descriptions for the same service w.r.t. the impact that the new knowledge will have on its knowledge base, and therefore the ability to minimize the costs by choosing to incorporate the least expensive alternative.

The current framework is susceptible to optimisations. As stated in Section 3, grouping is based on the assumption that an overlap in the signature S means that the involved *axioms* will entail useful knowledge when considered together. While this would not happen if they were to be placed in different groups, the above assumption does not always hold, as it is possible that some of the *axioms* in a group may not participate in any inference, or the inferences they participate into can be drawn from a different set of *axioms*. Thus, groups may be larger than necessary, which can reduce their utility in terms of size and expressivity reduction for impact evaluation.

An alternative approach that does not cause the groups to grow more than necessary might be based on the work presented in [22]; Kalyanpur et al present an algorithm to find all explanations for a specific entailment. They apply this algorithm to entailments of a specific group, as it is possible to verify which axioms are actually necessary and which axioms can be left out of the group. Also, it is possible to devise strategies based on the number of explanations in which an axiom is involved in order to decide whether a specific axiom may be removed from a group, e.g., an axiom A participating only in an explanation for axiom B, which also has another explanation not involving A, may be removed without causing B to be retracted. However, the extra computational cost must be taken into account; Kalyanpur et al envisage the use of their algorithm in user oriented tools, where the user explicitly asks for explanations; the cost involved in using the same algorithm automatically over a large number of axioms is intuitively very large w.r.t. the cost of running it on a reduced number of axioms. Further investigation is included in the future developments of this work.

7 Conclusions

This paper has introduced a framework for OWL knowledge base change management through grouping, useful in an open environment to help an agent to make a rational choice when confronted with the possibility of changing its knowledge base. The paper presented a real world example, based on a fragment of the Galen ontology, to show the feasibility of choosing rationally, using the impact function, whether a specific change tot he knowledge base may be accepted.

An evaluation of the validity of using groups, in terms of the ability to restrict the impact of a change to a subset of the knowledge base with a lower DL expressivity than the whole knowledge, and therefore with lower computational complexity, has been presented as well. An initial evaluation of the properties of a group has been introduced, discussing the kind of entailed axioms that are not handled by the current definition of group.

Ongoing developments are centred on optimisation of grouping in order to reuse reasoning results between groups with overlapping sets of *axioms*, which actually cause a duplication of the reasoning effort; also, the theoretical properties of a group will be explored to provide more formal results characterising the behavior of the framework.

References

1. Studer, R., Benjamins, V., Fensel, D.: Knowledge engineering, principles and methods. Data and Knowledge Engineering **25** (1998) 161–197
2. Singh, M., Huhns, M.: Service-Oriented Computing: Semantics, Processes, Agents. Wiley (2005)
3. Euzenat, J., Shvaiko, P.: Ontology Matching. Springer-Verlag, Heidelberg (DE) (2007)
4. ISWC'07 International Workshop on Emergent Semantics and Ontology Evolution. Volume 292., CEUR (2007)
5. Laera, L., Tamma, V., Euzenat, J., Bench-Capon, T., Payne, T.R.: Reaching Agreement over Ontology Alignments. In: 5th International Semantic Web Conference (ISWC 2006), Springer (2006) 371–384
6. Aberer, K., Cudr-Mauroux, P., Ouksel, A.M., Catarci, T., Hacid, M.S., Illarramendi, A., Kashyap, V., Mecella, M., Mena, E., Neuhold, E.J., et al.: Emergent semantics principles and issues. In: Proceedings of Database Systems for Advances Applications, 9th International Conference, DASFAA 2004. (2004)

7. Noy, N.F., Klein, M.C.A.: Ontology evolution: Not the same as schema evolution. Knowledge Information Systems **6** (2004) 428–440
8. Fensel, D., van Harmelen, F.: Unifying reasoning and search to web scale. IEEE Internet Computing **11** (2007) 9695
9. Payne, T.R.: Web services from an agent perspective. IEEE Intelligent Systems **23** (2008) 12–14
10. Sensoy, M., Yolum, P.: A cooperation-based approach for evolution of service ontologies. In Padgham, Parkes, Muller, Parsons, eds.: Proc. of 7th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2008), May, 12-16, 2008, Estoril, Portugal. (2008) 837–844
11. Kalyanpur, A., Parsia, B., Sirin, E., Grau, B.C.: Repairing unsatisfiable concepts in owl ontologies. In Sure, Y., Domingue, J., eds.: The Semantic Web: Research and Applications, 3rd European Semantic Web Conference, ESWC 2006, Budva, Montenegro, June 11-14, 2006, Proceedings. Volume 4011 of Lecture Notes in Computer Science., Springer (2006) 170–184
12. Simon, H.: Models of Bounded Rationality. Volume 3. MIT Press (1997)
13. Baader, F., Calvanese, D., McGuinness, D., Nardi, D., Patel-Schneider, P., eds.: The Description Logic Handbook. Cambridge University Press (2003)
14. Carroll, J.J., Dickinson, I., Dollin, C., Reynolds, D., Seaborne, A., Wilkinson, K.: Jena: implementing the semantic web recommendations. In Feldman, S.I., Uretsky, M., Najork, M., Wills, C.E., eds.: Proceedings of the 13th international conference on World Wide Web - Alternate Track Papers & Posters, WWW 2004, New York, NY, USA, May 17-20, 2004, ACM (2004) 74–83
15. SPARQL: Query language for RDF. (W3C Recommendation, 15th January 2008)
16. Sirin, E., Parsia, B., Grau, B.C., Kalyanpur, A., Katz, Y.: Pellet: A practical OWL-DL reasoner. Journal of Web Semantics **5** (2007)
17. Grau, B.C., Parsia, B., Sirin, E., Kalyanpur, A.: Modularity and web ontologies. In Doherty, P., Mylopoulos, J., Welty, C.A., eds.: Proceedings, Tenth International Conference on Principles of Knowledge Representation and Reasoning, Lake District of the United Kingdom, June 2-5, 2006, AAAI Press (2006) 198–209
18. Pitts, A.M.: On an interpretation of second order quantification in first order intuitionistic propositional logic. J. Symb. Log. **57** (1992) 33–52
19. Wolter, F.: Fusions of modal logics revisited. In Kracht, M., de Rijke, M., Wansing, H., Zakharyaschev, M., eds.: Advances in Modal Logic. CSLI, Stanford, CA (1998)
20. Schlobach, S., Huang, Z., Cornet, R., van Harmelen, F.: Debugging Incoherent Terminologies. Journal of Automated Reasoning **3** (2007) 317–349
21. Halashek-Wiener, C., Parsia, B., Sirin, E.: Description logic reasoning with syntactic updates. In Meersman, R., Tari, Z., (et al.), eds.: Ontologies, Databases and Applications of Semantics (ODBASE) 2006 International Conference. Volume 4275 of Lecture Notes in Computer Science., Springer (2006) 722–737
22. Kalyanpur, A., Parsia, B., Horridge, M., Sirin, E.: Finding all justifications of owl dl entailments. In Aberer, K., Choi, K.S., Noy, N.F., (et al.), eds.: The Semantic Web, 6th International Semantic Web Conference, 2nd Asian Semantic Web Conference, ISWC 2007 + ASWC 2007, Busan, Korea, November 11-15, 2007. Volume 4825 of Lecture Notes in Computer Science., Springer (2007) 267–280